Department of Mathematics and Statistics

Exact penalty methods for nonlinear optimal control problems

Eunice Anita Blanchard

This thesis is presented for the Degree of
Doctor of Philosophy
of
Curtin University

November 2014

Declaration

I affirm that the material in this thesis is the result of my own original research and has not been submitted for any other degree, diploma, or award.

....

Eunice Anita Blanchard

November 17, 2014

Abstract

In this thesis, we develop computational methods for solving three classes of non-standard optimal control problems:

(i) optimal control problems with discontinuous objective functions arising in aquaculture operations;

(ii) impulsive optimal control problems with minimum subsystem durations; and

(iii) optimal control problems involving dual-mode hybrid systems with state-dependent switching conditions.

The first problem involves a dynamic model of shrimp farming in which shrimp are harvested at several intermediate times during the production cycle. The problem is to choose the optimal harvesting times and corresponding optimal harvesting fractions to maximize the total revenue. The main difficulty with this problem is that the selling price of shrimp is a piecewise constant function of the average shrimp weight. Consequently, the revenue function is discontinuous. By performing a time-scaling transformation and introducing a set of binary variables, the shrimp harvesting problem is converted into an equivalent optimization problem that has a smooth objective function. An exact penalty method is then used to solve this equivalent problem.

The second class of problems considered in this thesis involves impulsive switched systems. Such systems consist of multiple subsystems operating in succession, with possible instantaneous state jumps occurring when the system switches from one subsystem to another. The control variables are the subsystem durations and a set of system parameters influencing the state jumps. We do not require that every subsystem must be active during the time horizon. To the best of our knowledge, existing publications on the control of impulsive switched systems do not consider this situation. It may be optimal to "delete" certain subsystems, especially when the optimal number of switches is unknown. However, any active subsystem must be active for a minimum non-negligible duration of time. This restriction leads to a disconnected region for the subsystem durations. The problem of choosing the subsystem durations and the system parameters to minimize a given cost function is a non-standard optimal control problem that cannot be solved using conventional techniques. A computational algorithm is developed by combining a time-scaling transformation and an exact penalty method for solving this problem. The effectiveness of this algorithm is demonstrated by considering a numerical example on the optimization of shrimp harvesting operations.

The third problem addressed in this thesis involves optimizing a class of hybrid systems whose state dynamics switch between two distinct modes. For this class of systems, the times at which the mode transitions occur cannot be specified directly, but are instead governed by a state-dependent switching condition. Various types of constraints (including constraints that depend on two or more fixed time points called characteristic times as well as all-time state constraints) are imposed on the system which further increase the complexity of the problem. Moreover, the control variables consist of a set of continuous-time input signals. By introducing an auxiliary binary-valued control function to represent the current mode, the hybrid system under consideration is transformed into a standard dynamic system subject to path constraints. The computational algorithm developed involves the control parameterization technique, a time-scaling transformation and an exact penalty method, for determining suboptimal input signals for the system. A numerical example on cancer chemotherapy is included to demonstrate the effectiveness of our proposed algorithm.

Publications

The following research papers were completed during my PhD candidature.

- E. A. Blanchard, R. Loxton and V. Rehbock, "A computational algorithm for a class of non–smooth optimal control problems arising in aquaculture operations" *Journal of Applied Mathematics and Computation*, vol. 219, no. 16, pp. 8738–8746, 2013.
- E. Blanchard, R. Loxton and V. Rehbock, "Optimal control of impulsive switched systems with minimum subsystem durations", *Journal of Global Optimization*, Accepted for publication.
- E. A. Blanchard, R. Loxton and V. Rehbock, "Dynamic optimization of dual–mode hybrid systems with state–dependent switching conditions", *Optimization Letters*, Submitted.

Acknowledgements

The research reported in this thesis was carried out between February 2011 and June 2014, while I was a PhD student in the Department of Mathematics and Statistics, Curtin University.

I wish to convey my heartfelt thanks to my supervisors, Dr. Ryan Loxton and A/Prof. Volker Rehbock for enabling me to reach this milestone of completing my PhD thesis. More specifically, I am grateful for their patience and diligence in guiding my research, despite the fact that I was going through a challenging set of circumstances in my personal life, throughout my entire study period at Curtin University.

As I returned to the academic arena in 2011 after a break of many long years, I had to learn to cope with the vast change in technology and computing that had taken place during those years. Dr. Ryan Loxton recognized this fact and gave me his valuable time from November 2010 - February 2011 to update my skills on the Fortran Programming and Latex. He mentored me during those summer months, freely giving up his time, although I was not enrolled at Curtin as an official doctoral candidate nor did I have certainty then, that I would obtain the CUPSA scholarship in time to come. I wish to thank Dr. Ryan Loxton sincerely for his willingness and his enthusiasm to work with me prior to my official enrolment in the PhD programme. This prior learning proved to be a great stepping stone and boosted my confidence to undertake this study. Furthermore, I am very grateful to Dr. Ryan Loxton for the times he has given up his time willingly, on week–ends particularly when I had a deadline to meet in relation to my research work.

Dr. Volker Rehbock worked part–time and was the co–ordinator for Engineering Mathematics during my period of study at Curtin University. In addition, he had the responsibility to supervise numerous PhD candidates of Mathematics. Amidst all these responsibilities and demands on his time, he guided and mentored me with his technical skills as well as his wealth of research experience. In addition, he made time to supervise my work, especially during the periods in which Dr. Ryan Loxton was away in China. I am grateful to Dr. Volker Rehbock for the flexibility and sensitivity he has shown towards my personal life challenges as I tried to cope with them, alongside with my role as a PhD student. In his quiet and calm manner, he would encourage me to keep going with my research work, whenever I felt over-whelmed or diffident about the challenges that I was facing in my personal life.

I wish to thank Prof. Yong Hong Wu, the postgraduate coordinator in the Department of Mathematics and Statistics, for helping me on many occasions with postgraduate administrative issues. He was also the chairman of my thesis committee.

I have known Prof. Kok Lay Teo since my son Jonathan Blanchard became an undergraduate student of this department in February 2006. I contacted Prof. Teo in August 2010 and told him that I would like to do a PhD at Curtin University. He then organised for me to meet with a few staff members of the department so that I could make a firm decision about the topic that I would undertake to study. I am greatly indebted to him for all the help and support I received from him in the past few years as well as for the exceptional way in which he mentored and guided my son, Jonathan during the four years of his undergraduate study at Curtin University.

I wish to express my gratitude to Prof. Lou Caccetta (Head of the Department of Mathematics and Statistics) and Dr. Greg Gamble for their support to me in various ways during the past 3 – 4 years.

I wish to convey my special thanks Dr. Kyle Chow, Dr. Changjun Yu and Dr. Bin Li, for your friendship, help and support in various ways during the past few years. I would like to thank my fellow PhD students especially those students who shared Room 444 with me, for your friendship, kindness and particularly for being supportive of me during my happy and sad moments, in the past few years.

I take this opportunity to thank the general staff of the Department of Mathematics and Statistics, especially Shuie Liu, Cheryl Cheng and Jeanie Darmago for going out of your way to help me and providing me with professional help on numerous occasions.

Above all, I give thanks and glory to Jesus Christ who has sustained me through each and every day of my study period and provided all of the people and material resources necessary to complete my PhD candidature.

Every good and perfect gift is from above – James 1:17.

Contents

CHAPTER 1

Introduction

1.1 Historical Perspective of Optimal Control

What is optimal control? It is a subject in applied mathematics dealing with the problem of finding a control law for a given dynamical system such that a certain optimality criterion is achieved. The concept of *Calculus of Variations* serves as the foundational groundwork for the theory of optimal control. In real life, one finds that individuals, organizations or groups of people need to make optimum decisions when a large number of factors are changing simultaneously, while adhering to many constraints and limitations imposed on the system. For example, a manufacturing company needs to make decisions to increase its profits, while having a limit on the cost of raw materials, factoring in the cost of labour (part of which could be a fixed cost with the rest being a variable cost depending on the hours worked), seasonal factors, etc. In the shrimp farming industry, the management would like to maximize the total profit by harvesting the shrimp at optimum times during the production cycle while factoring in the growth of the shrimp, the cost of harvesting and the various prices that different size shrimp would fetch subject to limitations on the capacity of the shrimp growing pond.

The origin of the foundational notion of optimal control dates back to the 1600's. In June 1696, Johann Bernoulli, who lived in Groningen, Netherlands, invited the mathematicians of that era to solve the Brachystochrone problem. What is the Brachystochrone problem? It can be described as: *Find the shape of the curve down which a bead sliding from rest and accelerated by gravity will slip (without friction) from one point to another in the shortest time.* The term "Brachystochrone" is derived from the Greek words "brachistos" (which means the shortest) and "chronos" (which means time). Six mathematicians, including Newton and Leibniz submitted their solutions to this problem. It would be fair to say that a foundation (which subsequently led to the concept of "calculus of variations") was laid in 1697, when Johann Bernoulli published his solution to the Brachystochrone problem [109].

Intense activity on similar kinds of problems followed among the mathematicians of that time. Euler, being a student of Bernoulli and Lagrange, became interested in these kinds of problems, which then led to their joint research in this area. The concept of calculus of variations began when Lagrange was studying the "Tautochrone Problem". The calculus of variations is concerned with the maxima or minima (collectively called extrema) of functionals. A functional has another function as its argument, somewhat analogous to the way a function has a numerical variable as its argument, and thus a functional can be described as a function of a function. Functionals have extrema with respect to the elements in a function space, defined over a given domain.

In his letters to Euler between 1754 and 1756, Lagrange shared his observation of a connection between minimizing functionals and finding extrema of a function which led to the creation of the *Euler–Lagrange Equations*: solving these equations leads to a function for which a given functional is stationary. These are regarded as the fundamental equations of the calculus of variations. They state that if J is defined by an integral of the form:

$$J = \int_a^b f\left(t, x(t), \dot{x}(t)\right) dt,$$

where $\dot{x} = \frac{d}{dt}\{x\}$, then J has a stationary value if the following Euler–Lagrange differential equation is satisfied.

$$\frac{\partial f}{\partial x} - \frac{\partial}{\partial t}\left\{\frac{\partial f}{\partial \dot{x}}\right\} = 0.$$

As time progressed, other mathematicians such as, Legendre (1786), Brunacci (1810), Hamilton, (1827), Dirichlet (1829), Gauss (1829), Poisson (1831), Jocobi, (1837), Cauchy (1844), Strauch (1849), Hesse (1857), Clebsch (1858), Riemann (1859) and Carll (1885) became contributors to the calculus of variations. However, it was Karl Weierstrass who made very significant advancements in the field during 1800's. He established a necessary condition for the existence of strong extrema of variational problems, in addition to his input into creating the Weierstrass–Erdmann condition, which gives sufficient conditions for an extremal to have a corner along a given extrema, and allows one to find a minimizing curve for a given integral.

Dirichlet considered the problem of finding a function which solves a specified partial differential equation (PDE) in the interior of a given region and which takes prescribed values on the boundary of the region. Dirichlet's principle assumes that within every bounded connected domain, at least one function can be found to reduce a given integral to a minimum. Since Dirichlet's integral is bounded from below, Riemann (who coined the term Dirichlet's principle) took for granted the existence of an infimum. However, Karl Weierstrass in 1870, gave an example of a functional that does not attain its minimum. Subsequently, David Hilbert in 1900 effectively restored Dirichlet's principle by showing that the existence of a solution in this instance depended delicately on the smoothness of the boundary and the prescribed data.

In its classical period, the calculus of variations depended on many of the pivotal theorems in the theory of differential equations. Early 20th century contributors to the research and development of the calculus of variations include David Hilbert (1900), Henri Lebesgue (1904), Oskar Bolza (1914), Leonida Tonelli (1923), Jacques Hadamard (he introduced the *method of descent* in 1923 as a method for solving a partial differential equation in several real or complex variables) and Marston Morse (1931).

In 1900, David Hilbert outlined 23 mathematical problems to the International Congress of Mathematicians in Paris. The nineteenth problem asked whether the solutions of regular problems in the calculus of variations were always necessarily analytic. Sergei Bernstein (1904) showed that solutions of nonlinear elliptic analytic equations in two variables are analytic. Bernstein's result was improved over the years by several authors, such as Petrowsky (1939), who reduced the differentiability requirements on the solution needed to prove that it is analytic. The direct methods in the calculus of variations showed the existence of solutions, with very weak differentiability properties. For many years there was a gap between these results. The solutions that could be constructed were known to have square integrable second derivatives, which was not quite strong enough to prove they were analytic. In addition, continuity of the first derivatives was needed. This gap was filled independently by Ennio De Giorgi (1957), and John Forbes Nash (1958). They were able to show that the solutions had first derivatives which were *Hölder continuous*, which, by previous results, implied that the solutions are analytic whenever the differential equation has analytic coefficients. This completed the solution of Hilbert's nineteenth problem.

In 1956, the Russian mathematician Lev Semenovich Pontryagin and his students formulated the famous *Pontryagin's maximum (or minimum) principle* (PMP), which is fundamental to the modern theory of optimization. This principle is used in optimal control theory (which is an extension of the calculus of variations) to find the best possible control for taking a dynamical system from one state to another, especially in the presence of constraints on the state or input controls. PMP is still widely used in current times to solve optimal control problems [39, 108]. Pontryagin also introduced the idea of a bang–bang principle, to describe situations where a maximum "steer" should be applied to a system. He lost his eyesight due to a stove explosion when he was 14. Despite his blindness, Pontryagin was able to become one of the greatest mathematicians of the 20th century.

1954 marked the birth of the concept of *Dynamic Programming*. This was due to work of the US–born applied mathematician Richard Bellman, who created the theory for treating the mathematical problems arising from the study of various multi–stage decision processes. He established the "Principle of Optimality": an optimal policy has the property that whatever the initial state and initial decision are, the remaining decisions must constitute an optimal policy with regard to the state resulting from the first decision [2, 3]. He proposed the *Bellman Equation* that applies to discrete–time dynamic optimization problems. Bellman showed that a dynamic optimization problem in discrete time can be stated in a recursive, step–by–step form by writing down the relationship between the value function at two consecutive periods. This relationship (between the two value functions) is called the Bellman equation. In continuous–time dynamic optimization problems, the analogous equation is a partial differential equation known as the *Hamilton–Jacobi–Bellman Equation*.

Bellman's remarkable research and publications in the field of applied mathematics continued for the next two decades despite the fact that he was diagnosed with a brain tumour in 1973, which left him almost a total cripple. He realised that the conceptual, analytic and computational aspects of the mathematical model must be considered simultaneously [4] and was quick to embrace the advantages of using a digital computer. Bellman won many honours for his outstanding contributions in applied mathematics. The 1979 IEEE Awards Reception Brochure states: *Richard Bellman is a towering figure among the contributors to modern control theory and systems analysis. His invention of dynamic programming marked the beginning of a new era in the analysis and optimization of large–scale systems and opened a way for the application of sophisticated computer–oriented techniques in a wide variety of problem–areas ranging from the design of guidance systems for space vehicles to pest control and network optimization.*

As many real–life problems involve functions that are not linear, *nonlinear programming* became a popular area of research with the advent of fast digital computers. Nonlinear programming is the process of solving an optimization problem defined by a system of equalities and inequalities, (collectively termed as constraints), over a set of unknown real variables, along with an objective function to be maximized or minimized. In this case, certain constraints and/or the objective function are nonlinear. Reference [42] focuses on the subject of nonlinear programming in its early stages, while [41] gives a historical overview of it up to the mid 1970's. Many of the nonlinear programming problems at that time were solved using gradient–based methods (see [29, 47]). Subsequently, non–gradient methods were also developed. These include decomposition methods, approximation methods, discretization techniques and evolutionary algorithms.

Ralph Tyrrell Rockafellar (who completed his postgraduate degree at Harvard University in 1963) made significant contributions towards the development of optimization methodology as well as to convex analysis. Rockafellar's theory differs from classical analysis in that differentiability assumptions are replaced by convexity assumptions [87, 88]. He has continued his research work in the field for almost 5 decades (see [74, 87–92]), [74, 92] being two of his more recent publications.

Klaus Schittkowski has reviewed and analyzed many optimization methods over three decades and developed various methods for optimizing different classes of problems, since 1975. Schittkowski has done extensive research in nonlinear programming. His early works included [96] and [97], which present solution techniques to nonlinear optimization problems. He has developed a widely used optimization subroutine called NLPQL [98, 99] to solve a very general class of nonlinear programming problems. References [100–102] are some of Schittkowski's most recent research and publications in the area of optimization methods.

In the 1980's, Frank H. Clarke authored the books titled *Optimization and Non–smooth Analysis* and *Methods of Dynamic and Non–smooth Optimization* in addition to many joint research papers with other collaborators of that era. Commercial availability of digital computers at this point of time spurred on the research on numerical solution methods for optimal control problems.

Kok Lay Teo made outstanding contributions to the progress of the research and development of optimization methods to various classes of optimal control problems over the last two and half decades [24, 110, 111, 113, 119, 122, 124]. Due to his innovative ability for finding solution techniques to optimal control problems, Teo (together with other collaborators) produced an optimal control software called MISER in the 1990's. The authors in [120] use MISER3 to solve various problems. This software was fine–tuned in 2004 (see [34]) to be used in conjunction with a few additional inputs from the user, for optimal control problems involving discontinuities in the state variables. Today, this software package serves as an excellent tool for broadening the scope of the research in the area of optimal control and numerical analysis.

Researchers and developers of numerical algorithms prior to the 1990's should be commended for their pioneering work in this field, which laid the foundations for the wave of research in this area as computing speed increased rapidly after that era. In the subsequent years, the advancement of computer technology enhanced the research potential of numerical techniques and thus, novel algorithms were developed to solve complex optimal control problems. Reference [81] highlights the fact that the first algorithms for optimal control were aimed at unconstrained problems, while [125] discusses breakthroughs in the development of efficient computational methods for optimal control of switched systems in the late nineties to early 2000's. This survey paper primarily focuses on the issue of discretization and discusses the relative merits and limitations of discretization methods and non–discretization methods.

As a result of the research undertaken (globally) over the last three hundred and fifty years, advancements in the concept of optimal control have been achieved through the deeper understanding and knowledge of the calculus of variations, optimality conditions, dynamic programming and nonlinear programming. Moreover, the solution methods have undergone several changes over this period from analytical methods to sophisticated numerical techniques that require the use of an optimal control software or other computer programs. Over the last three decades, many optimal control softwares have come into existence to implement a variety of different solution techniques. As a result of all these developments, we are able to efficiently solve many types of unconstrained and constrained optimal control problems.

Currently, researchers are striving to solve more intricate optimal control problems such as problems with discontinuities in the objective [5], optimal control problems involving hybrid systems [79, 128, 135] and optimal control problems involving switched systems [20, 25, 37, 53, 66, 117, 123] and problems with non–smooth objective [14], to name just a few. Furthermore, sophisticated numerical techniques form part of the solution methods for optimal control problems in recent times. They include various forms of transformations, [5, 6], approximations [14, 65], introduction of binary variables [6], constraint transcriptions [50,58,123], modifying a constrained optimal control problem into a sequence of unconstrained optimal control problems [39], filled function methods [121,127], gradient–based optimization methods [15, 16] and penalty methods [49, 123, 129–131]. In addition, many researchers have developed convergence analysis for certain solution methods to optimal control problems [64, 132], which provides the mathematical foundation and authenticity for the methods in question.

In summary, the developments of optimal control theory and its applications continue to advance into the 21st century, through the work of mathematicians, numerical analysts and engineers. The mathematicians focus on the characterization of theoretical properties of optimization problems and convergence properties for optimization algorithms, while the numerical analysts develop various optimization methods that are efficient from a practical consideration. They are concerned with ease of computations, numerical stability and performance. Generally, the engineers are focused on the application of optimization methods to real problems while checking the reliability, robustness and efficiency of such methods.

1.2 Mathematical Formulation of an Optimal Control Problem

In an optimal control problem, one chooses an admissible control for a dynamical system such that an objective function (i.e., a performance index) is optimized. A general optimal control problem can be described as follows.

We consider a dynamical system given by:

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t), t), \qquad t \in [0, T], \tag{1.1}$$

and initial condition

$$\boldsymbol{x}(0) = \boldsymbol{x}^0, \tag{1.2}$$

where

- $\boldsymbol{u}(t) \in \mathbb{R}^{n_u}$ is the control vector at time t (a time–varying decision variable);
- $\boldsymbol{x}(t) \in \mathbb{R}^{n_x}$ is the state vector at time t;
- $\boldsymbol{f}$ is a given function, assumed to be continuously differentiable with respect to $\boldsymbol{x}$ and $\boldsymbol{u}$, and piecewise continuous with respect to time t;
- T is the terminal time;
- n_x is the number of states; and
- n_u is the number of controls.

Note that from a practical perspective, the control strategy for (1.1) and (1.2) is subject to lower and upper limits and cannot be completely arbitrary. Hence, one often defines a control restraint set U as:

$$U = \left\{ [u_1, \ldots, u_{n_u}]^\top \in \mathbb{R}^{n_u} : \alpha_i \leq u_i \leq \beta_i,\ i = 1, \ldots, n_u \right\},$$

where α_i and β_i are given constants such that $\alpha_i < \beta_i$ for each $i = 1, \ldots, n_u$. A bounded measurable function $\boldsymbol{u} : [0, T] \to \mathbb{R}^{n_u}$ such that $\boldsymbol{u}(t) \in U$ for all $t \in [0, T]$ is called an admissible control. Let $\mathcal{U}$ be the class of all such admissible controls.

The state is influenced by the control through system (1.1). Note that the control can change its value during the time interval $[0, T]$. Hence, for a given control, the state evolves according to the system of ordinary differential equations (1.1) with initial condition

(1.2) over the time horizon $[0, T]$. Let $\boldsymbol{x}(\cdot|\boldsymbol{u})$ denote the solution of (1.1) and (1.2) corresponding to $\boldsymbol{u} \in \mathcal{U}$.

Many practical problems involve a variety of different constraints imposed on the state and control. A canonical form for general constraints imposed on the system can be mathematically expressed as:

$$G_i(\boldsymbol{u}) = \Phi_i(\boldsymbol{x}(T|\boldsymbol{u})) + \int_0^T \mathcal{L}_i\left(\boldsymbol{x}(t|\boldsymbol{u}), \boldsymbol{u}(t), t\right) dt \begin{cases} = 0, & i = 1, \ldots, q_e, \\ \geq 0, & i = q_e + 1, \ldots, q. \end{cases} \tag{1.3}$$

In an optimal control problem, we seek to minimize a cost functional of the form

$$G_0(\boldsymbol{u}) = \Phi_0(\boldsymbol{x}(T|\boldsymbol{u})) + \int_0^T \mathcal{L}_0\left(\boldsymbol{x}(t|\boldsymbol{u}), \boldsymbol{u}(t), t\right) dt \tag{1.4}$$

subject to the dynamics given by (1.1), the initial condition (1.2) and the canonical constraints given by (1.3). Here, Φ_i and $\mathcal{L}_i, i = 0, 1, \ldots, q$, are given continuously differentiable functions. We refer to this problem as Problem 1A.

1.3 Numerical Techniques for Optimal Control Problems

Numerical methods for solving optimal control problems can be divided into two major categories, namely the indirect methods and the direct methods. Reference [83] gives a good overview of the two types of methods and discusses the various approaches within each category. The shooting method and multiple shooting method [26, 78, 94] are the most common examples of indirect methods. The famous Pontryagin's Minimum (or Maximum) principle [82] forms the underlying basis for these methods. This principle supplies the first order necessary conditions for optimality via a two point boundary value problem, which (in principle) can be solved using shooting methods to obtain a numerical solution.

Direct methods (see, for example, [28]) are fundamentally different from indirect methods. They involve approximating the control (or both the control and the state) of the optimal control problem in some appropriate manner. This discretization process [95] is necessary to transfer the continuous models and equations into discrete counterparts, and thus make them suitable for numerical computation. In the case where only the control is approximated, the direct method is called a control parameterization method [111]. In the case where both the state and the control are approximated, the direct method is called a state discretization method [27,38,103]. The control parameterization enhancing technique was developed in [45] to improve the accuracy of the control parameterization technique. This aspect is demonstrated in [120] through the application of these techniques to a time–delayed optimal control problem. Reference [85] surveys developments in the control parameterization method during the 1990's and reference [52,126] gives an updated overview of this method. Furthermore, the control parameterization [57] and the control parameterization enhancing technique have been used in [18,43,51,58,120] to solve optimal control problems arising in various applications. The control parameterization method and its applications are discussed in more detail in the next section.

1.3.1 Control Parameterization Method

The first step in the control parameterization method is to define a partition of the time horizon $[0, T]$ of Problem 1A as follows:

$$P = \{t_0, t_1, \ldots, t_N\},$$

where $t_0 = 0$, $t_N = T$ and $t_{j-1} < t_j$ for each $j = 1, \ldots, N$. The set P defines a set of knots, which could be unequally spaced. The next step is to approximate the control by a linear combination of basis functions, i.e.,

$$\boldsymbol{u}(t) = \sum_{j=1}^{N} \boldsymbol{\sigma}_j \psi_j(t),$$

where each ψ_j is a basis function corresponding to the knot point t_j. In most applications of control parameterization, the basis functions $\psi_j, j = 1, \ldots, N$, are zero order or first order splines with finite support. However, the basis functions could also take the form of quadratic spline functions or even cubic spline functions. For more details on the basis functions, see Chapter 2 of [34]. Note that the coefficients $\boldsymbol{\sigma}_j$ of the basis functions are decision variables which need to be optimally chosen.

When the basis functions are piecewise constant [34], we can define $\psi_j(t) = \chi_{[t_{j-1}, t_j)}(t)$, an indicator function defined by

$$\chi_{[t_{j-1}, t_j)}(t) = \begin{cases} 1, & \text{if } t \in [t_{j-1}, t_j), \\ 0, & \text{otherwise.} \end{cases}$$

Then the approximate control function can be written as:

$$\boldsymbol{u}^N(t) = \sum_{j=1}^{N} \boldsymbol{\sigma}_j \chi_{[t_{j-1}, t_j)}(t), \tag{1.5}$$

where

$$\boldsymbol{\sigma}_j \in U, \quad j = 1, \ldots, N. \tag{1.6}$$

The heights of the approximating functions are decision variables, also known as control parameters. See Figure 1.1 for an example.

Let $\boldsymbol{\sigma} = [(\boldsymbol{\sigma}_1)^\top, \ldots, (\boldsymbol{\sigma}_N)^\top]^\top$. Furthermore, let Σ be the set of all vectors $\boldsymbol{\sigma}$ such that their components satisfy (1.6). Now, substituting (1.5) into (1.1) yields:

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{\sigma}_j, t), \qquad t \in [t_{j-1}, t_j), \quad j = 1, \ldots, N. \tag{1.7}$$

Let $\boldsymbol{x}^N(\cdot|\boldsymbol{\sigma})$ denote the solution of (1.7) and (1.2) corresponding to $\boldsymbol{\sigma} \in \Sigma$.

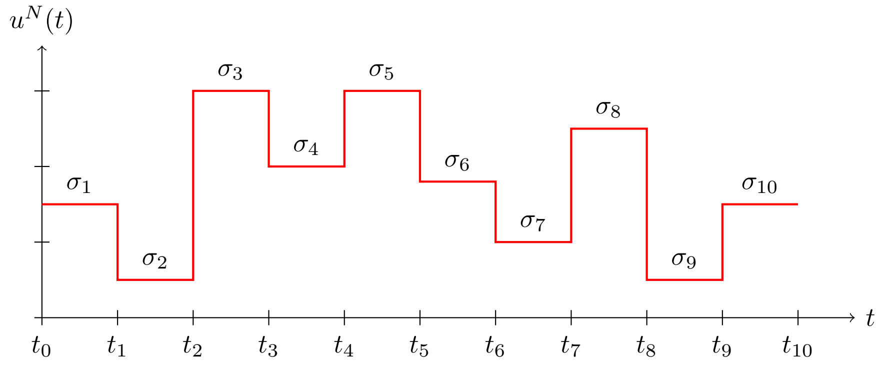

Figure 1.1: An example of piecewise constant control approximation with equidistant knot points ($N = 10$ subintervals).

Therefore, our transformed objective function now becomes:

$$G_0^N(\boldsymbol{\sigma}) = \Phi_0(\boldsymbol{x}^N(T|\boldsymbol{\sigma})) + \sum_{j=1}^{N} \int_{t_{j-1}}^{t_j} \mathcal{L}_0\left(\boldsymbol{x}^N(t|\boldsymbol{\sigma}), \boldsymbol{\sigma}_j, t\right) dt. \tag{1.8}$$

Likewise, the constraints (1.3) are transformed to:

$$G_i^N(\boldsymbol{\sigma}) = \Phi_i(\boldsymbol{x}^N(T|\boldsymbol{\sigma})) + \sum_{j=1}^{N} \int_{t_{j-1}}^{t_j} \mathcal{L}_i\left(\boldsymbol{x}^N(t|\boldsymbol{\sigma}), \boldsymbol{\sigma}_j, t\right) dt \begin{cases} = 0, & i = 1, \ldots, q_e, \\ \geq 0, & i = q_e + 1, \ldots, q. \end{cases} \tag{1.9}$$

We now formulate an approximate problem as follows: Choose $\boldsymbol{\sigma} \in \Sigma$ to minimize the objective given by (1.8) subject to the system (1.7), the initial condition (1.2) and the canonical constraints (1.9). We refer to this problem as Problem 1B. Clearly, in the approximation technique used to obtain Problem 1B, logical reasoning would dictate that a finer partition (i.e. a larger value of N) would produce a more accurate solution. This is proven by the convergence results in [64].

Problem 1B can be viewed as a mathematical programming problem in which a finite number of decision variables need to be optimized. To solve this problem using numerical optimization techniques, the gradients (i.e. the partial derivatives) of the cost and constraint functions (which are both in the same canonical form) with respect to each $\boldsymbol{\sigma}_j$ are required. The relevant gradient formulae are given in the next section of this thesis. With these gradient formulae, Problem 1B can be solved using a gradient–based optimization method such as sequential quadratic programming [17, 70, 77, 102].

1.3.2 The Optimal Control Software MISER

The optimal control software MISER was originally developed by K.L. Teo and C.J. Goh in 1988. The first version was not user friendly, nor was it able to handle continuous inequality constraints imposed on the optimal control system. Therefore, MISER3, with MISER3.3 [35] being the latest version, was developed. It is a great tool for solving modern optimal control problems. The theoretical basis for the development of MISER3.3

is the concept of control parameterization, which was discussed in the previous subsection. MISER3.3 aims to solve various classes of optimal control problems (see [5, 6, 14, 59, 119, 131]).

The main virtue of MISER3.3 is that it can be used to solve a wide range of optimal control problems with multiple constraints and jump conditions on the state variables. The software caters for three standard forms of constraints: canonical constraints in the form of (1.3), continuous inequality constraints that restrict the control and state values at all times and linear constraints involving only the control variables. Note that the linear constraints depend only on the control and hence have gradient formulae that can be computed with ease.

MISER3.3 is designed for users with a basic knowledge of multi–variable calculus and fundamental FORTRAN programming skills. This knowledge is sufficient to solve basic optimal control problems. For more complex problems that do not fit the standard framework of the MISER3.3 software, a deeper understanding of the internal workings of the software is required to transform such problems into a solvable form.

To explain the technical elements of MISER3.3, we formulate a generalization of Problem 1A considered earlier. Let us consider the following system dynamics:

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{z}, t), \qquad t \in [0, T], \tag{1.10}$$

with the initial condition

$$\boldsymbol{x}(0) = \boldsymbol{x}^0(\boldsymbol{z}), \tag{1.11}$$

where

- $\boldsymbol{u}(t) \in \mathbb{R}^{n_u}$ is the control vector at time t (a time–varying decision variable);
- $\boldsymbol{x}(t) \in \mathbb{R}^{n_x}$ is the state vector at time t;
- $\boldsymbol{z} \in \mathbb{R}^{n_z}$ is the system parameter vector (a time–invariant decision variable);
- $\boldsymbol{f}$ is a given function, assumed to be continuously differentiable with respect to $\boldsymbol{x}$ and $\boldsymbol{u}$, and piecewise continuous with respect to time t;
- $\boldsymbol{x}^0(\boldsymbol{z})$ is a given function, (defining the initial state vector) assumed to be continuously differentiable with respect to $\boldsymbol{z}$;
- T is the terminal time;
- n_x is the number of states;
- n_u is the number of controls; and
- n_z is the number of system parameters.

As some optimal control problems have dynamics which involve state jumps at certain times, we impose jump conditions as follows:

$$\boldsymbol{x}(\tau_k^+) = \boldsymbol{x}(\tau_k^-) + \boldsymbol{\phi}^k(\boldsymbol{x}(\tau_k^-), \boldsymbol{z}), \qquad k = 1, \ldots, m-1, \tag{1.12}$$

where $\tau_k, k = 1, \ldots, m-1$, are given jump times and $\boldsymbol{\phi}^k : \mathbb{R}^{n_x} \times \mathbb{R}^{n_z} \to \mathbb{R}^{n_x},\ k = 1, \ldots, m-1$, are continuously differentiable given functions.

The latest version of MISER (i.e. MISER 3.3 developed in 2004) incorporates the facility for state jumps as described in (1.12). This was a significant advancement in MISER since the previous versions could only handle problems with continuous state trajectories. The optimal control problems formulated in [5, 6, 119] contained state jumps and were solved successfully using MISER3.3.

Let $\boldsymbol{x}(\cdot|\boldsymbol{u}, \boldsymbol{z})$ be the solution of (1.10)-(1.12) corresponding to the control $\boldsymbol{u} \in \mathcal{U}$ and the system parameter vector $\boldsymbol{z} = [z_1, \ldots, z_{n_z}]^\top \in \mathcal{Z}$, where $\mathcal{Z}$ is a compact convex subset that describes the feasible range for the system parameter vector.

Consider the following canonical constraints imposed on the system:

$$G_i(\boldsymbol{u}, \boldsymbol{z}) = \sum_{k=1}^{m} \Phi_{i,k}(\boldsymbol{x}(\tau_k|\boldsymbol{u}, \boldsymbol{z}), \boldsymbol{z}) + \int_0^T \mathcal{L}_i\left(\boldsymbol{x}(t|\boldsymbol{u}, \boldsymbol{z}), \boldsymbol{u}(t), \boldsymbol{z}, t\right) dt \begin{cases} = 0, & i = 1, \ldots, q_e, \\ \geq 0, & i = q_e + 1, \ldots, q, \end{cases} \tag{1.13}$$

where $\Phi_{i,k} : \mathbb{R}^{n_x} \times \mathbb{R}^{n_z} \to \mathbb{R}$, $i = 1, \ldots, q,\ k = 1, \ldots, m$, and $\mathcal{L}_i : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \times \mathbb{R}^{n_z} \times \mathbb{R} \to \mathbb{R}, i = 1, \ldots, q$, are given continuously differentiable functions. This form of canonical constraint depends on the state values at multiple characteristic times $\tau_1, \ldots, \tau_m$, where $0 = \tau_0 < \tau_1 < \cdots < \tau_m = T$. Thus, it can be considered as a more general form of the canonical constraints (1.3).

In addition, we consider continuous inequality constraints imposed on the system, which can be expressed mathematically by:

$$g_i(\boldsymbol{x}(t|\boldsymbol{u}, \boldsymbol{z}), \boldsymbol{z}) \geq 0, \qquad t \in [0, T], \qquad i = 1, \ldots, n_c, \tag{1.14}$$

where $g_i : \mathbb{R}^{n_x} \times \mathbb{R}^{n_z} \to \mathbb{R},\ i = 1, \ldots, n_c$, are given continuously differentiable functions.

We define a cost function in the same form as the canonical functions in (1.13):

$$G_0(\boldsymbol{u}, \boldsymbol{z}) = \sum_{k=1}^{m} \Phi_{0,k}(\boldsymbol{x}(\tau_k|\boldsymbol{u}, \boldsymbol{z}), \boldsymbol{z}) + \int_0^T \mathcal{L}_0\left(\boldsymbol{x}(t|\boldsymbol{u}, \boldsymbol{z}), \boldsymbol{u}(t), \boldsymbol{z}, t\right) dt, \tag{1.15}$$

where $\Phi_{0,k} : \mathbb{R}^{n_x} \times \mathbb{R}^{n_z} \to \mathbb{R}$, $k = 1, \ldots, m$, and $\mathcal{L}_0 : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \times \mathbb{R}^{n_z} \times \mathbb{R} \to \mathbb{R}$ are given continuously differentiable functions.

Our problem is to choose an admissible pair $(\boldsymbol{u}, \boldsymbol{z}) \in \mathcal{U} \times \mathcal{Z}$ to minimize (1.15) subject to the system dynamics (1.10)-(1.12), the canonical constraints (1.13) and the continuous inequality constraints (1.14). This is a general formulation for the class of problems that MISER3.3 can solve.

MISER3.3 uses the concept of control parameterization to solve the class of optimal control problems described above. Hence, using the approximate control function (1.5) defined in the previous section, the system dynamics in (1.10) can be transformed to:

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{\sigma}_j, \boldsymbol{z}, t), \qquad t \in [t_{j-1}, t_j), \quad j = 1, \ldots, N. \tag{1.16}$$

Let $\boldsymbol{x}^N(\cdot|\boldsymbol{\sigma}, \boldsymbol{z})$ be the solution of the system defined by (1.16) and (1.11)-(1.12), corresponding to $(\boldsymbol{\sigma}, \boldsymbol{z}) \in \Sigma \times \mathcal{Z}$.

Using the control parameterization technique described in the previous section, we transform the cost function (1.15) to:

$$G_0^N(\boldsymbol{\sigma}, \boldsymbol{z}) = \sum_{k=1}^{m} \Phi_{0,k}(\boldsymbol{x}^N(\tau_k|\boldsymbol{\sigma}, \boldsymbol{z}), \boldsymbol{z}) + \sum_{j=1}^{N} \int_{t_{j-1}}^{t_j} \mathcal{L}_0\left(\boldsymbol{x}^N(t|\boldsymbol{\sigma}, \boldsymbol{z}), \boldsymbol{\sigma}_j, \boldsymbol{z}, t\right) dt. \tag{1.17}$$

Similarly, the canonical constraints in (1.13) can be transformed to:

$$\begin{aligned} G_i^N(\boldsymbol{\sigma}, \boldsymbol{z}) \;=\; & \sum_{k=1}^{m} \Phi_{i,k}(\boldsymbol{x}^N(\tau_k|\boldsymbol{\sigma}, \boldsymbol{z}), \boldsymbol{z}) \\ & + \sum_{j=1}^{N} \int_{t_{j-1}}^{t_j} \mathcal{L}_i\left(\boldsymbol{x}^N(t|\boldsymbol{\sigma}, \boldsymbol{z}), \boldsymbol{\sigma}_j, \boldsymbol{z}, t\right) dt \begin{cases} = 0, & i = 1, \ldots, q_e, \\ \geq 0, & i = q_e + 1, \ldots, q. \end{cases} \end{aligned} \tag{1.18}$$

Furthermore, the continuous inequality constraints in (1.14) become:

$$g_i(\boldsymbol{x}^N(t|\boldsymbol{\sigma}, \boldsymbol{z}), \boldsymbol{z}) \geq 0, \qquad t \in [0, T], \qquad i = 1, \ldots, n_c. \tag{1.19}$$

Our approximate problem, which MISER3.3 formulates automatically, is to choose an admissible pair $(\boldsymbol{\sigma}, \boldsymbol{z}) \in \Sigma \times \mathcal{Z}$ to minimize (1.17) subject to the system dynamics (1.16), the initial condition (1.11), the jump conditions (1.12), the canonical constraints (1.18) and the continuous inequality constraints (1.19). We refer to this problem as Problem 1C. MISER3.3 solves Problem 1C using nonlinear programming techniques.

For nonlinear programming problems, most of the popular numerical solution methods require the analytical gradients of the cost function as well as the constraint functions. However, in dynamic optimization problems such as Problem 1C, the analytical gradients cannot be easily obtained. Thankfully, MISER can automatically compute the gradients of the canonical functions with respect to the control values and the system parameters. The gradient formulae MISER uses are stated below. Derivations of these gradients are given in the MISER manual [34].

For each $i = 0, 1, \ldots, q$, let the corresponding Hamiltonian H_i be defined by:

$$H_i(\boldsymbol{x}, \boldsymbol{\sigma}, \boldsymbol{z}, t, \boldsymbol{\lambda}^i) = \mathcal{L}_i(\boldsymbol{x}, \boldsymbol{\sigma}, \boldsymbol{z}, t) + \{\boldsymbol{\lambda}^i\}^\top \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\sigma}, \boldsymbol{z}, t). \tag{1.20}$$

Here, the costate variable $\boldsymbol{\lambda}^i(t)$, satisfies the following differential equation:

$$\frac{d\boldsymbol{\lambda}^i(t)}{dt} = -\left[\frac{\partial H_i(\boldsymbol{x}^N(t|\boldsymbol{\sigma},\boldsymbol{z}),\boldsymbol{\sigma}_j,\boldsymbol{z},t,\boldsymbol{\lambda}^i(t))}{\partial \boldsymbol{x}}\right]^\top, \quad t \in (t_{j-1},t_j), \quad j = 1,\dots,N, \tag{1.21}$$

with the terminal condition

$$\boldsymbol{\lambda}^i(\tau_m) = \left[\frac{\partial \Phi_{i,m}(\boldsymbol{x}^N(\tau_m|\boldsymbol{\sigma},\boldsymbol{z}),\boldsymbol{z})}{\partial \boldsymbol{x}}\right]^\top, \tag{1.22}$$

and the jump conditions

$$\boldsymbol{\lambda}^i(\tau_k^-)^\top = \boldsymbol{\lambda}^i(\tau_k^+)^\top\left[\frac{\partial \boldsymbol{\phi}^k(\boldsymbol{x}^N(\tau_k^-|\boldsymbol{\sigma},\boldsymbol{z}),\boldsymbol{z})}{\partial \boldsymbol{x}}\right] + \left[\frac{\partial \Phi_{i,k}(\boldsymbol{x}^N(\tau_k^-|\boldsymbol{\sigma},\boldsymbol{z}),\boldsymbol{z})}{\partial \boldsymbol{x}}\right]. \tag{1.23}$$

Let $\boldsymbol{\lambda}^i(\cdot|\boldsymbol{\sigma},\boldsymbol{z})$ be the solution of the costate system (1.21), (1.22) and (1.23). The gradient of the i^{th} canonical function with respect to the control values can be expressed in terms of the costate as follows:

$$\frac{\partial G_i^N(\boldsymbol{\sigma},\boldsymbol{z})}{\partial \boldsymbol{\sigma}_j} = \int_{t_{j-1}}^{t_j}\left[\frac{\partial H_i(\boldsymbol{x}^N(t|\boldsymbol{\sigma},\boldsymbol{z}),\boldsymbol{\sigma}_j,\boldsymbol{z},t,\boldsymbol{\lambda}^i(t))}{\partial \boldsymbol{\sigma}}\right]dt. \tag{1.24}$$

Furthermore, the gradient of the i^{th} canonical function with respect to the system parameters is given by:

$$\begin{aligned}\frac{\partial G_i^N(\boldsymbol{\sigma},\boldsymbol{z})}{\partial \boldsymbol{z}} = \sum_{k=1}^{m}\frac{\partial \Phi_{i,k}(\boldsymbol{x}^N(\tau_k|\boldsymbol{\sigma},\boldsymbol{z}),\boldsymbol{z})}{\partial \boldsymbol{z}} &+ \boldsymbol{\lambda}^i(\tau_0^+|\boldsymbol{\sigma},\boldsymbol{z})^\top\frac{\partial \boldsymbol{x}^0(\boldsymbol{z})}{\partial \boldsymbol{z}}\\ &+\sum_{k=1}^{m-1}\boldsymbol{\lambda}^i(\tau_k^-|\boldsymbol{\sigma},\boldsymbol{z})^\top\frac{\partial \boldsymbol{\phi}^k(\boldsymbol{x}^N(\tau_k^-|\boldsymbol{\sigma},\boldsymbol{z}),\boldsymbol{z})}{\partial \boldsymbol{z}}\\ &+\sum_{j=1}^{N}\int_{t_{j-1}}^{t_j}\frac{\partial H_i(\boldsymbol{x}^N(t|\boldsymbol{\sigma},\boldsymbol{z}),\boldsymbol{\sigma}_j,\boldsymbol{z},t,\boldsymbol{\lambda}^i(t))}{\partial \boldsymbol{z}}dt.\end{aligned} \tag{1.25}$$

On the basis of the formulae (1.24) and (1.25), we present the following algorithm for computing the gradients of the canonical functions in Problem 1C:

1. For each given pair $(\boldsymbol{\sigma},\boldsymbol{z}) \in \Sigma \times \mathcal{Z}$, compute the solution $\boldsymbol{x}^N(\cdot|\boldsymbol{\sigma},\boldsymbol{z})$ of the state system by solving the differential equations (1.16) together with the initial condition (1.11) and the jump conditions (1.12) forward in time from $t = 0$ to $t = T$.

2. Using the solution from *Step 1*, compute the corresponding values of $G_i(\boldsymbol{\sigma},\boldsymbol{z})$ from (1.18) for $i = 0, 1, \dots, q$.

3. Using the solution from *Step 1*, compute the solution $\boldsymbol{\lambda}^i(\cdot|\boldsymbol{\sigma},\boldsymbol{z})$ of the costate differential equation (1.21) with the conditions (1.22) and (1.23) backward in time from $t = \tau_m$ to $t = 0$.

4. Using the results from *Steps 2* and *3*, compute the gradients of $G_i(\boldsymbol{\sigma},\boldsymbol{z})$, $i =$

$0, 1, \ldots, q$, with respect to $\boldsymbol{\sigma}$ and the system parameter $\boldsymbol{z}$ from (1.24) and (1.25).

The continuous inequality constraints given in (1.14) are distinctly different from the canonical constraints. In terms of the underlying mathematical programming problem, they represent infinite dimensional constraints. Therefore, gradient–based optimization methods cannot handle these constraints directly, and hence one needs to apply the so–called *constraint transcription technique* to transform them into a solvable form. The constraint transcription technique is a powerful tool for the solution of optimal control problems. It can be used to cast complex forms of constraints into a neat canonical form that can be handled by standard optimal control software, such as MISER. Moreover, this technique enables us to considerably reduce the number of constraints imposed on the system (when necessary), thus easing the computational challenges posed by optimal control problems with continuous inequality constraints.

The concept of constraint transcription was initially proposed in [110] and [112], where it was employed to handle continuous inequality constraints that only involve the state variables. It was extended in [35] to handle optimal control problems subject to continuous inequality constraints on the state as well as on the control. Further development of the transcription method is seen in [56, 84, 111]) to address various forms of continuous inequality constraints. Constraint transcription techniques can be effectively applied in conjunction with smoothing technique [123], control parameterization [58], with the time–scaling transformation [50,122] or with the penalty approach [49,84]. Reference [85] surveys control parameterization methods and control parameterization enhancing methods prior to 2000, where the constraint transcription technique has been employed in optimal control problems subject to continuous inequality constraints on the state and/or control.

To run the MISER program, a user–specified data file needs to be created. This file specifies problem data such as the number of states, number of controls, number of system parameters, the total number of constraints (divided into 6 different categories), details of the knot sets and bounds for the controls. In addition, the program requires the user to give an initial value (i.e. an educated guess) for the controls as well as for each of the system parameters. The user may also specify error tolerances for the numerical integration of the state and costate equations. Similarly, convergence criteria can be set for the mathematical programming routine used to solve the approximate problem, including a tolerance on how closely the constraints should be satisfied. Thus, the MISER program allows great flexibility to incorporate a broad range of optimal control models and to tailor the solution to user specified requirements.

When using MISER3.3, the following factors must be taken into consideration.

- Jump points for the state variables may coincide with some of the characteristic times in the constraints.
- The SQP routine used within MISER3.3 only guarantees a locally optimal solution. Hence, the results obtained depend on the initial guesses for the various decision parameters, i.e., different initial guesses may result in different local solutions.

- The solution satisfies all convergence criteria to the prescribed accuracy (as specified in the data file), provided that there is no error message to indicate non–convergence. Note that one of MISER's output parameters indicates whether convergence was successful.
- When endeavouring to solve non–standard optimal control problems, users must use their ingenuity to transform their problem into a standard form that can be readily solved using MISER3.3.

1.3.3 Time–scaling Technique

The time–scaling transformation technique was originally developed in [46] almost 15 years ago. The literature on this technique is now vast [5,6,51,122], thus demonstrating that the time–scaling technique has been one of the most popular tools in recent years for solving nonlinear optimal control problems. The main idea of the time–scaling transformation is to allow the switching points for the approximate piecewise constant control to be decision variables. The resulting approximate problem is transformed into an equivalent problem with fixed switching times, to avoid numerical difficulties associated with variable switching times. See [63] for a discussion of these difficulties.

Suppose that the switching times $t_j, j = 1, \ldots, N-1$, are decision variables. Then, $t_j, j = 1, \ldots, N$, satisfy the following ordering constraints:

$$0 = t_0 \leq t_1 \leq t_2 \leq \cdots \leq t_N = T. \tag{1.26}$$

We introduce a new time variable $s \in [0, N]$ and relate it to t through the following differential equation:

$$\dot{t}(s) = \mu(s), \qquad t(0) = 0, \tag{1.27a}$$

and

$$t(N) = T, \tag{1.27b}$$

where $\mu : [0, N] \to \mathbb{R}$ is a piecewise constant function satisfying the bounds

$$0 \leq \mu(s) \leq T, \qquad s \in [0, N]. \tag{1.27c}$$

To ensure that the characteristic times are transformed appropriately, we assume that each characteristic time coincides with one of the switching times: $t_{\nu_k} = \tau_k$, where $\nu_k \in \{1, \ldots, N\}$. Then we require the following constraints in addition to (1.27b):

$$t(\nu_k) = \tau_k, \quad k = 1, \ldots, m, \tag{1.27d}$$

where ν_k is an integer such that $\nu_k \in \{1, \ldots, N\}$.

Let

$$\theta_j = t_j - t_{j-1}, \qquad j = 1, \ldots, N. \tag{1.28}$$

We can now express $\mu(s)$ mathematically as follows:

$$\mu(s) = \sum_{j=1}^{N} \theta_j \chi_{[j-1,j)}(s), \tag{1.29}$$

where $\chi_{[j-1,j)}(s)$ is the indicator function defined by

$$\chi_{[j-1,\ j)}(s) = \begin{cases} 1, & \text{if } s \in [j-1, j), \\ 0, & \text{otherwise.} \end{cases}$$

Thus, the new control function $\mu(s)$ has fixed switching times at $s = 1, \ldots, N$. For $s \in [k-1,\ k]$, it follows from equations (1.27) and (1.28) that

$$t(s) = \int_0^s \mu(\eta)d\eta = \sum_{j=1}^{k-1} \theta_j + \theta_k(s-k+1) \tag{1.30a}$$

and thus

$$t(k) = \sum_{j=1}^{k} \theta_j = \sum_{j=1}^{k} (t_j - t_{j-1}). \tag{1.30b}$$

Let Θ be the set of all $\boldsymbol{\theta} = [\theta_1, \theta_2, \ldots, \theta_N]^\top \in \mathbb{R}^N$ satisfying $\theta_j \geq 0, j = 1, \ldots, N$.

Let $\widetilde{x}(s) = x(t(s))$. Under the time–scaling transformation, the dynamics of Problem 1C become:

$$\dot{\widetilde{\boldsymbol{x}}}(s) = \boldsymbol{f}(\widetilde{\boldsymbol{x}}(s), \boldsymbol{\sigma}_j, \boldsymbol{z}, t(s)), \qquad s \in [j-1, j), \quad j = 1, \ldots, N, \tag{1.31}$$

with the jump conditions

$$\widetilde{\boldsymbol{x}}({\nu_k}^+) = \widetilde{\boldsymbol{x}}({\nu_k}^-) + \boldsymbol{\phi}^j(\widetilde{\boldsymbol{x}}({\nu_k}^-), \boldsymbol{z}), \qquad k = 1, \ldots, m-1, \tag{1.32}$$

and the initial conditions

$$\widetilde{\boldsymbol{x}}(0) = \widetilde{\boldsymbol{x}}^0(\boldsymbol{z}). \tag{1.33}$$

Let $\widetilde{\boldsymbol{x}}^N(\cdot|\boldsymbol{\sigma}, \boldsymbol{z}, \boldsymbol{\theta})$ be the solution of (1.31) and (1.33) corresponding to $(\boldsymbol{\sigma}, \boldsymbol{z}, \boldsymbol{\theta}) \in \Sigma \times \mathcal{Z} \times \Theta$. Here, the original time variable t is treated as a state variable satisfying (1.27).

The canonical constraints in (1.18) are then transformed to:

$$\begin{aligned} \widetilde{G}_i^N(\boldsymbol{\sigma}, \boldsymbol{z}, \boldsymbol{\theta}) &= \sum_{k=1}^{m} \Phi_{i,k}(\widetilde{\boldsymbol{x}}^N(\nu_k|\boldsymbol{\sigma}, \boldsymbol{z}, \boldsymbol{\theta}), \boldsymbol{z}) \\ &+ \sum_{j=1}^{N} \int_{j-1}^{j} \theta_j \mathcal{L}_i\left(\widetilde{\boldsymbol{x}}^N(s|\boldsymbol{\sigma}, \boldsymbol{z}, \boldsymbol{\theta}), \boldsymbol{\sigma}_j, \boldsymbol{z}, t(s)\right) ds \begin{cases} = 0, & i = 1, \ldots, q_e, \\ \geq 0, & i = q_e + 1, \ldots, q. \end{cases} \end{aligned} \tag{1.34}$$

Furthermore, the continuous inequality constraints (1.19) are transformed to:

$$g_i(\widetilde{\boldsymbol{x}}^N(s|\boldsymbol{\sigma}, \boldsymbol{z}, \boldsymbol{\theta}), \boldsymbol{z}) \geq 0, \quad s \in [j-1, j), \quad j = 1, \ldots, N, \quad i = 1, \ldots, n_c. \tag{1.35}$$

Problem 1C is thus transformed into the following problem in the new time horizon:

Choose $(\boldsymbol{\sigma}, \boldsymbol{z}, \boldsymbol{\theta}) \in \Sigma \times \mathcal{Z} \times \Theta$ to minimize the transformed cost function

$$\widetilde{G}_0^N(\boldsymbol{\sigma}, \boldsymbol{z}, \boldsymbol{\theta}) = \sum_{k=1}^{m} \Phi_{0,k}(\widetilde{\boldsymbol{x}}^N(\nu_k|\boldsymbol{\sigma}, \boldsymbol{z}, \boldsymbol{\theta}), \boldsymbol{z}) + \sum_{j=1}^{N} \int_{j-1}^{j} \theta_j \mathcal{L}_0 \left(\widetilde{\boldsymbol{x}}^N(s|\boldsymbol{\sigma}, \boldsymbol{z}, \boldsymbol{\theta}), \boldsymbol{\sigma}_j, \boldsymbol{z}, t(s)\right) ds \tag{1.36}$$

subject to the system represented by (1.31), the initial condition (1.33), the canonical constraints (1.34), the continuous inequality constraints (1.35) and the jump conditions (1.32). We refer to this problem as Problem 1D.

1.3.4 Exact Penalty Method

Penalty approaches to solving nonlinear optimization problems have been in existence for over three decades [9, 19, 32, 104]. Recent applications of penalty methods are found in [5, 6, 14, 49, 123]. Such methods are used to convert a constrained optimization problem into an unconstrained problem, which is much easier to solve than a constrained problem. The general idea of penalty methods is to append a suitable function of the constraints (which measures the constraint violation) to the objective. For some penalty functions, it can be shown that the local optimizers of the penalized problem are precisely the optimizers of the original problem. Such methods are called *exact penalty methods*. For a general discussion of penalty function methods, see the book by W. Huyer and A. Neumaier [32] and the references cited therein.

Recently, the penalty methods developed in [129–131] have become powerful tools in the solution techniques of nonlinear optimization and optimal control problems. To deal with the continuous infinite–dimensional constraints, an early transcription method was proposed in [110]. This transcription method was extended into a penalty method in [84]. More recently, the exact penalty method in [55, 130] was developed to handle such constraints. Reference [132] investigates the convergence analysis for the exact penalty function in an optimal control problem with infinite–dimensional constraints. To address the issue of discrete decision variables, the exact penalty methods in [129, 131] can be utilised.

Consider the following class of mathematical programming problems with infinite–dimensional constraints: Minimize the cost function $G_0(\boldsymbol{y}, \boldsymbol{v})$ subject to the constraints:

$$G_i(\boldsymbol{y}, \boldsymbol{v}) \begin{cases} = 0, & i = 1, \ldots, q_e, \\ \geq 0, & i = q_e + 1, \ldots, q, \end{cases} \tag{1.37}$$

and

$$g_i(t, \boldsymbol{y}, \boldsymbol{v}) \geq 0, \qquad t \in [a, b], \qquad i = 1, \ldots, n_c. \tag{1.38}$$

Here,

- $\boldsymbol{y} \in \mathbb{R}^{n_y}$ is a vector of continuous–valued decision variables;
- $\boldsymbol{v} \in \{0, 1\}^{n_v}$ is a vector of binary–valued decision variables;

- $G_i : \mathbb{R}^{n_y} \times \{0,1\}^{n_v} \to \mathbb{R}$, $i = 0, 1, \ldots, q$, are given functions representing the cost and conventional constraints, assumed to be continuously differentiable for each $\boldsymbol{v} \in \{0,1\}^{n_v}$;
- $g_i : \mathbb{R} \times \mathbb{R}^{n_y} \times \{0,1\}^{n_v} \to \mathbb{R}$, $i = 1, \ldots, n_c$, are given functions representing the infinite–dimensional constraints, assumed to be continuously differentiable for each $\boldsymbol{v} \in \{0,1\}^{n_v}$.

We refer to this problem as Problem 1E. Such problems are known as semi–infinite programming problems in the literature [12, 55, 105, 130].

Note that Problem 1D in the previous section (and more generalised versions of this problem involving discrete–valued decision variables) can be viewed as a special case of this general form, where the continuous inequality constraints (1.35) correspond to the infinite–dimensional constraints (1.38). The two key challenges for solving Problem 1E are the binary–valued variables and the infinite–dimensional constraints. Standard gradient–based methods in optimization cannot deal directly with these aspects, and thus one way of overcoming these challenges is to apply penalty methods. Since the exact penalty method (along with the time–scaling method and control parameterization technique) is the basis for the computational methods proposed in this thesis, we now give a discussion below on the exact penalty method for solving Problem 1E.

Let $\boldsymbol{v} = [v_1, v_2, \ldots, v_{n_v}]^\top$. We relax the binary variables by regarding them as continuous–valued decision variables while imposing the following additional constraints:

$$v_i(1 - v_i) \leq 0, \quad i = 1, \ldots, n_v, \tag{1.39a}$$

and

$$0 \leq v_i \leq 1, \quad i = 1, \ldots, n_v. \tag{1.39b}$$

Consequently, we have the following equivalent problem: Minimize the cost function $G_0(\boldsymbol{y}, \boldsymbol{v})$ subject to the constraints:

$$G_i(\boldsymbol{y}, \boldsymbol{v}) \begin{cases} = 0, & i = 1, \ldots, q_e, \\ \geq 0, & i = q_e + 1, \ldots, q, \end{cases} \tag{1.40}$$

and

$$g_i(t, \boldsymbol{y}, \boldsymbol{v}) \geq 0, \qquad t \in [a, b], \qquad i = 1, \ldots, n_c, \tag{1.41}$$

$$h(v_i) = v_i(v_i - 1) \geq 0, \quad i = 1, \ldots, n_v, \tag{1.42a}$$

and the bounds

$$0 \leq v_i \leq 1, \quad i = 1, \ldots, n_v. \tag{1.42b}$$

This problem is referred to as Problem 1F. In Problem 1F, the v_i's are regarded as continuous–valued variables. Standard optimization algorithms will struggle to handle constraints (1.42a), as they define a disconnected region. Therefore, using the strategy introduced in [129–131], an exact penalty function $\widehat{G}_\delta(\boldsymbol{y}, \boldsymbol{v}, \epsilon)$ is constructed as follows:

$$\widehat{G}_\delta(\boldsymbol{y}, \boldsymbol{v}, \epsilon) = \begin{cases} G_0(\boldsymbol{y}, \boldsymbol{v}), & \text{if } \epsilon = 0,\ \Delta(\boldsymbol{y}, \boldsymbol{v}) = 0, \\ G_0(\boldsymbol{y}, \boldsymbol{v}) + \epsilon^{-\lambda}\Delta(\boldsymbol{y}, \boldsymbol{v}) + \delta\epsilon^{\gamma}, & \text{if } \epsilon > 0, \\ +\infty, & \text{otherwise,} \end{cases}$$

where

- ϵ is a new decision variable;
- $\delta > 0$ is the penalty parameter;
- λ and γ are positive constants satisfying $1 \leq \gamma \leq \lambda$; and
- $\Delta(\boldsymbol{y}, \boldsymbol{v})$ is a function measuring the *constraint violation*.

The mathematical expression for the constraint violation is given by:

$$\Delta(\boldsymbol{y}, \boldsymbol{v}) = \sum_{i=1}^{q_e} \left[G_i(\boldsymbol{y}, \boldsymbol{v})\right]^2 + \sum_{i=q_e+1}^{q} \left[\min\left\{0, G_i(\boldsymbol{y}, \boldsymbol{v})\right\}\right]^2$$
$$+ \sum_{i=1}^{n_c} \int_a^b \left[\min\left\{0, g_i(t, \boldsymbol{y}, \boldsymbol{v})\right\}\right]^2 dt + \sum_{i=1}^{n_v} \left[\min\left\{0, h_i(v_i)\right\}\right]^2.$$

The new decision variable ϵ is subject to the following bounds:

$$0 \leq \epsilon \leq \widetilde{\epsilon}, \tag{1.43}$$

where $\widetilde{\epsilon} > 0$ is a small positive number.

We now define the following unconstrained problem: Choose $(\boldsymbol{y}, \boldsymbol{v}) \in \mathbb{R}^{n_y} \times [0,1]^{n_v}$ and $\epsilon \in [0, \widetilde{\epsilon}]$ to minimize $\widehat{G}_\delta(\boldsymbol{y}, \boldsymbol{v}, \epsilon)$ subject to the bounds (1.42b). We refer to this problem as Problem 1G.

Now consider Problem 1D in the previous section of this thesis. Using the same analogy which was used to transform the constrained optimization problem (i.e. Problem 1E) into an unconstrained problem (i.e. Problem 1G), we transform Problem 1D in the following manner. This transformation is necessary to handle the continuous inequality constraints in Problem D. In the ensuing sections of this thesis, we will also consider generalizations of Problem 1D that contain discrete variables.

The *constraint violation* term for Problem 1D is:

$$\Delta(\boldsymbol{\sigma}, \boldsymbol{z}, \boldsymbol{\theta}) = \sum_{i=1}^{q_e} \left[\widetilde{G}_i^N(\boldsymbol{\sigma}, \boldsymbol{z}, \boldsymbol{\theta})\right]^2 + \sum_{i=q_e+1}^{q} \left[\min\left\{0, \widetilde{G}_i^N(\boldsymbol{\sigma}, \boldsymbol{z}, \boldsymbol{\theta})\right\}\right]^2$$
$$+ \sum_{i=1}^{n_c} \int_0^N \left[\min\left\{0, g_i(\widetilde{\boldsymbol{x}}^N(s|\boldsymbol{\sigma}, \boldsymbol{z}, \boldsymbol{\theta}))\right\}\right]^2 ds + \sum_{k=1}^{m} \left[t(\nu_k) - \tau_k\right]^2 + \left[t(N) - T\right]^2.$$

Based on the constraint violation term above, we create a pseudo–objective function as below:

$$\widehat{J}_\delta(\boldsymbol{\sigma}, \boldsymbol{z}, \boldsymbol{\theta}, \epsilon) = \begin{cases} \widetilde{G}_0^N(\boldsymbol{\sigma}, \boldsymbol{z}, \boldsymbol{\theta}) & \text{if } \epsilon = 0, \ \Delta(\boldsymbol{\sigma}, \boldsymbol{z}, \boldsymbol{\theta}) = 0, \\ \widetilde{G}_0^N(\boldsymbol{\sigma}, \boldsymbol{z}, \boldsymbol{\theta}) + \epsilon^{-\lambda}\Delta(\boldsymbol{\sigma}, \boldsymbol{z}, \boldsymbol{\theta}) + \delta\epsilon^\gamma, & \text{if } \epsilon > 0, \\ +\infty, & \text{otherwise,} \end{cases}$$

where

- ϵ is a new decision variable;
- $\delta > 0$ is the penalty parameter;
- λ and γ are positive constants satisfying $1 \leq \gamma \leq \lambda$.

The new decision variable ϵ is subject to the following bounds:

$$0 \leq \epsilon \leq \widetilde{\epsilon}, \tag{1.44}$$

where $\widetilde{\epsilon} > 0$ is a small positive number.

We now define the following unconstrained problem: Choose $(\boldsymbol{\sigma}, \boldsymbol{z}, \boldsymbol{\theta}) \in \Sigma \times \mathcal{Z} \times \Theta$ and $\epsilon \in [0, \widetilde{\epsilon}]$ to minimize $\widehat{J}_\delta(\boldsymbol{\sigma}, \boldsymbol{z}, \boldsymbol{\theta}, \epsilon)$. We refer to this problem as Problem 1H.

Note that $\Delta(\boldsymbol{\sigma}, \boldsymbol{z}, \boldsymbol{\theta}) = 0$ if and only if the constraints in Problem 1D are satisfied. The convergence analysis in [132] shows that, for a sufficiently large penalty parameter δ, a local minimizer of the unconstrained optimization problem is a local minimizer of the original optimal control problem with canonical constraints and continuous inequality constraints. This key result justifies the use of the exact penalty function $\widehat{J}_\delta(\boldsymbol{\sigma}, \boldsymbol{z}, \boldsymbol{\theta}, \epsilon)$.

1.4 Synopsis of the Thesis

The purpose of this thesis is to present new computational methods for three different classes of optimal control problems, each of which is more complex than the class discussed in Section 1.2. The efficiency and applicability of the proposed methods are demonstrated through realistic numerical examples. Each of the solution methods combines various numerical techniques to transform a constrained optimal control problem into an unconstrained problem that is readily solved by MISER3.3. The numerical convergence obtained in each of the examples is sufficiently convincing to demonstrate the reliability of the proposed algorithms. Furthermore, graphical illustrations based on the numerical results are incorporated into the relevant chapters, to enhance the understanding of the reader.

In this chapter, Section 1.1 captures a historical perspective of optimal control over the last 350 years. In the ensuing subsection, various aspects of numerical techniques of optimal control problems are discussed. In particular, three fundamental numerical techniques that form the basis for the solution methods to the optimal control problems addressed in this thesis are discussed. These are control parameterization, time–scaling transformation and the exact penalty method. As the optimal control software MISER3.3 has been used to solve the numerical examples given in this thesis, the functional elements of MISER3.3 are described in Subsection 1.3.2, based on a general optimal control formulation. A mathematical representation of this problem, which involves constraints with characteristic times, state jumps and system parameters, is given for the reader to grasp the technical aspects of MISER3.3.

Chapter 2 deals with an applied problem that arises within the shrimp farming industry. It involves the dynamic model for partial shrimp harvesting, proposed by Yu and Leung in [133], over a single production cycle of T weeks. It has 2 state variables, i.e. the average weight of an individual shrimp at time t and the number of shrimp remaining at time t. The dynamics include jump conditions due to the fact that shrimp are harvested at intermediate points of time within the production cycle. The aim is to maximize the revenue function J subject to the state dynamics and the jump conditions. The revenue function is a function of the price of shrimp. Yu and Leung use a fixed price per kilogram of shrimp in [133]. However, the price function in our problem is modified to allow different prices being assigned to different weight ranges of shrimp. This aspect made the revenue function a discontinuous function of the state.

Chapter 3 deals with optimal control problems involving impulsive switched systems. It focuses on the case when not all subsystems need to be active and that it is possible to delete certain subsystems within the context of an optimal solution. A computational algorithm is developed, by combining the time–scaling transformation and an exact penalty method, for solving this problem. The effectiveness of this algorithm is demonstrated by considering a numerical example on the optimization of shrimp harvesting operations, where 7 of the 10 switches are deleted, thus resulting in 3 switches remaining in the final optimal solution.

Chapter 4 focuses on a computational approach for optimizing a class of hybrid systems whose state dynamics switch between two distinct modes. The hybrid system under consideration is transformed into a standard dynamic system subject to path constraints. The computational algorithm developed involves the control parameterization technique, a time–scaling transformation and an exact penalty method, for determining suboptimal input signals for the system. A numerical example on cancer chemotherapy is included to demonstrate the effectiveness of our proposed algorithm.

The last chapter contains a summary of the main contributions of the thesis as well as an outlook for future research work involving optimal control problems.

CHAPTER 2

A Computational Algorithm for a Class of Optimal Control Problems with Discontinuous Objective Functions

2.1 Introduction

In [133], Yu and Leung proposed a mathematical model for shrimp farming over a single batch production cycle. This model was subsequently used to demonstrate that harvesting shrimps at intermediate times before the final harvest (called partial harvesting) is an optimal strategy for maximizing the overall profit. The dynamic equations in [133] describe shrimp growth by a uniform density-dependent growth rate and shrimp mortality by way of exponential decay. Research into aquaculture population dynamics shows that this mathematical model is appropriate in the shrimp farming context [62, 115].

In this chapter, we again consider the model proposed by Yu and Leung in [133]. The dynamics in this model involve jumps in the state variables because there is a sudden change in the number of shrimps when an intermediate harvest takes place. This model has one major limitation: it assumes that the shrimp price is constant, when in fact the price of shrimp usually varies significantly with the average shrimp weight. We improve the model by introducing a more realistic price function that is suitable for a commercial environment. More specifically, we have incorporated a piecewise constant price function, which depends on the average shrimp weight, into the revenue function.

The problem of choosing the harvest times and the harvest fractions (i.e. the percentage of shrimp stock extracted) to maximize the total revenue is an optimal control problem in which the objective function is discontinuous and the dynamic system experiences state jumps at variable time points. Such optimal control problems are called impulsive optimal control problems in the literature, and they have been an active area of research over the past decade [37, 48, 59, 80, 122, 124]. To handle the variable jump points, we apply the time-scaling technique [53, 66], which involves mapping the variable jump times to fixed integers, thus yielding an equivalent problem in a new time horizon. This transformation is necessary as most standard optimal control algorithms for impulsive systems can only handle jump times that are fixed [53, 60, 122]. Although the time-scaling transformation eliminates the variability in the jump times, it does not eliminate the discontinuity in the objective function. Hence, inspired by the relaxation approaches in [55, 76, 129], we introduce new binary variables into the objective function, together with linear and quadratic constraints, to transform the objective function into a smooth function. The resulting optimization problem can be solved using an exact penalty method. This involves adding continuous penalty terms to the cost function to transform the original non-smooth optimal control problem into an approximate unconstrained problem. The penalty terms are zero at feasible points and positive at infeasible points. Prior research [55, 129, 131] indicates that any local minimizer of the unconstrained problem will be a local minimizer of the original problem when the penalty parameter is sufficiently large.

In this chapter, we illustrate the effectiveness of the proposed approach by applying it to the shrimp farming model in [133]. The time-scaling transformation and exact penalty approach result in a problem that can be readily solved using MISER 3.3 [34], which is an optimal control software based on the control parameterization technique [14, 36, 54]. The approach described in this chapter can also be readily extended to more general optimal control problems involving discontinuous objective functions. To the best of our knowledge, this is the first attempt at generating numerical solutions for such non-smooth optimal control problems.

2.2 Problem Formulation

We start with the dynamic model proposed by Yu and Leung [133] in which the biological mortality and growth processes of the shrimp stock are described by the following differential equations:

$$\dot{n}(t) = -mn(t), \qquad n(0) = n_0, \tag{2.1}$$

$$\dot{w}(t) = g\big(f(t), w(t), n(t), t\big), \qquad w(0) = w_0, \tag{2.2}$$

where

- t is the time in weeks;
- $w(t)$ is the average weight of an individual shrimp in grams at time t;
- $n(t)$ is the number of remaining shrimp at time t;
- m is a given constant representing the natural mortality rate of the shrimp;
- $f(t)$ is the feeding rate at time t;
- g is a given function that is differentiable with respect to each of its arguments;
- n_0 and w_0 are given initial conditions at $t = 0$.

Let $[0, T]$ denote the time horizon over which a single production cycle takes place (the final harvest occurs at time $t = T$).

Suppose that N harvests ($N - 1$ intermediate harvests and 1 final harvest) take place during the production cycle. Let $\tau_j \in [0, T]$ denote the time of the j^{th} harvest, with τ_N referring to the final harvest time. Then we have the following constraint:

$$0 = \tau_0 \le \tau_1 \le \tau_2 \le \cdots \le \tau_N = T. \tag{2.3}$$

Let ν_j denote the fraction of shrimp stock harvested at time τ_j. Then clearly,

$$0 \le \nu_j \le 1, \qquad j = 1, \ldots, N. \tag{2.4}$$

The state variables n and w are subject to the following jump conditions at each harvest time $t = \tau_j$:

$$n(\tau_j^+) - n(\tau_j^-) = -\nu_j n(\tau_j^-), \tag{2.5}$$

$$w(\tau_j^+) - w(\tau_j^-) = 0, \tag{2.6}$$

where, for a general function $h(t)$, we adopt the notation $h(\tau^\pm) = \lim_{t \to \tau^\pm} h(t)$.

Equation (2.5) asserts that the difference in the number of shrimps before and after the j^{th} harvest is equal to the number of shrimps harvested at time $t = \tau_j$. Equation (2.6) simply states that the average shrimp weight is unchanged by the j^{th} harvest. This assumes, of course, that when harvesting the shrimp, a uniform cross-section of the shrimp stock is extracted.

In the general dynamics (2.1) and (2.2) proposed by Yu and Leung in [133], the feeding rate $f(t)$ is also a decision variable to be chosen optimally, in addition to the harvesting fractions ν_j and harvesting times τ_j. Note though, that no specific example of this general form was actually proposed by Yu and Leung and their numerical results were based on a simpler form of the dynamics not involving the feeding rate. Following their lead, we also ignore the feeding rate $f(t)$ and consider only the harvesting fractions and the corresponding harvesting times as decision variables. However, note that the computational approach in this chapter can be easily extended to also optimize the feeding rate $f(t)$ using the control parameterization technique described in [14, 36, 54].

Yu and Leung proposed the following general model for the revenue obtained at the j^{th} harvest time [133]:

$$\text{Revenue} = R_j\{p\big(w(\tau_j)\big), w(\tau_j), n(\tau_j^-), \nu_j, c_j, h\}, \tag{2.7}$$

where

- c_j is the variable cost of the j^{th} harvest in dollars per kilogram;
- h is the fixed cost associated with each harvest;
- $p\big(w(\tau_j)\big)$ is the sale price of shrimp in dollars per kilogram (*as a function of the average weight of shrimp at the* j^{th} *harvest*);
- R_j is a given continuously differentiable function.

We adopt the following specific model suggested by Yu and Leung for the total revenue over the production cycle $[0, T]$:

$$J = \sum_{j=1}^{N} \left(10^{-3} \{ p\big(w(\tau_j)\big) - c_j \} w(\tau_j) n(\tau_j^-) \nu_j - h \right). \tag{2.8}$$

The numerical examples in [53, 133] consider the revenue function (2.8) for the simple case when the price function $p\big(w(\tau_j)\big)$ is a constant. More realistically, the sale price of shrimp is heavily dependent on the average weight of the shrimp. Thus, in this chapter, we consider a more appropriate price function in which different prices are assigned to different weight ranges. This piecewise constant price function is defined as follows:

$$p\big(w(\tau_j)\big) = \alpha_i, \qquad \beta_{i-1} \leq w(\tau_j) < \beta_i, \qquad i = 1, \ldots, L, \tag{2.9a}$$

where

- L is the number of different price levels;
- β_0 is the left end point of the lowest weight range;
- β_i is the right end point of the i^{th} weight range;
- α_i is the sale price of shrimp stock in dollars per kilogram when the average weight lies in the interval $[\beta_{i-1}, \beta_i)$.

We assume without loss of generality that

$$\beta_0 < \beta_1 < \beta_2 < \cdots < \beta_L \tag{2.9b}$$

and

$$\alpha_1 < \alpha_2 < \alpha_3 < \cdots < \alpha_L. \tag{2.9c}$$

From a practical point of view, the price function (2.9a) is far more realistic than those used in [53,133]. It is worth mentioning that [134] also considers the prize-size relationship of shrimp in a partial harvesting situation. [134] focuses on a network-flow approach for determining an efficient harvesting policy, whereas we focus on an optimal control approach for an impulsive system model in this chapter.

Following the lead of Yu and Leung [133], we consider the following specific dynamics for the state variables n and w:

$$\dot{n}(t) = -mn(t), \qquad n(0) = n_0, \tag{2.10}$$

$$\dot{w}(t) = a - bw(t)n(t), \qquad w(0) = w_0, \tag{2.11}$$

where m, a and b are given constants.

We now formulate an optimal control problem as follows:

Choose the harvesting fractions ν_j, $j = 1, \ldots, N$ and the corresponding harvesting times τ_j $j = 1, \ldots, N$ to maximize the revenue function defined by (2.8) and (2.9a) subject to the dynamics described by equations (2.10) and (2.11), the constraints given by (2.3) and (2.4) and the jump conditions given by (2.5) and (2.6). We refer to this problem as Problem 2A. We assume throughout this chapter that a solution exists for Problem 2A.

When $L = 1$ (i.e. there is only one price level), Problem 2A reduces to the shrimp harvesting problem considered in [53, 133]. In this reduced problem, the objective is smooth, and therefore the problem can be solved effectively using the impulsive control techniques discussed in [53, 60]. However, in this chapter, we are interested in the more difficult case when $L > 1$; that is, when there are distinct price levels. In this case, Problem 2A presents two major challenges for existing numerical solution methods:

- The jump conditions (2.5) and (2.6) occur at variable time points;
- The objective function is discontinuous and hence non-differentiable.

For these reasons, Problem 2A with $L > 1$ cannot be solved using standard optimal control software such as MISER 3.3. In this chapter we use a time-scaling transformation to handle the variable jump points and a smoothing transformation to handle the discontinuous objective function. Both of these transformations are described in the next section.

2.3 Problem Transformation

2.3.1 Application of the Time-Scaling Technique

Problem 2A is an optimal control problem in which the harvesting times are decision variables to be chosen optimally. The state variable n undergoes an instantaneous jump at each harvesting time. From a computational point of view, it is well known that variable jump times cause major difficulties for standard optimal control algorithms. Such algorithms require that the jump times be fixed, whereas in Problem 2A, the jump times are decision variables to be optimized. Thus, we adopt the time-scaling transformation described in [46, 53], which enables us to map the harvesting times to fixed points in a new time horizon.

We first introduce a new time variable $s \in [0, N]$ and relate s to t through the following differential equation:

$$\dot{t}(s) = u(s), \qquad t(0) = 0, \tag{2.12a}$$

and

$$t(N) = T, \tag{2.12b}$$

where $u : [0, N] \to \mathbb{R}$ is a piecewise constant function satisfying the bounds

$$0 \leq u(s) \leq T, \qquad s \in [0, N]. \tag{2.13}$$

Let

$$\theta_j = \tau_j - \tau_{j-1}, \qquad j = 1, \ldots, N. \tag{2.14}$$

That is, θ_j represents the duration between the $(j-1)^{th}$ and j^{th} harvest times. Thus,

$$\theta_j \geq 0, \qquad j = 1, \ldots, N.$$

We express $u(s)$ mathematically as follows:

$$u(s) = \sum_{j=1}^{N} \theta_j \chi_{[j-1,j)}(s), \tag{2.15}$$

where $\chi_{[j-1,j)}(s) : \mathbb{R} \to \mathbb{R}$ is the indicator function defined by

$$\chi_{[j-1,\ j)}(s) = \begin{cases} 1, & \text{if } s \in [j-1, j), \\ 0, & \text{otherwise.} \end{cases}$$

Thus, the control function $u(s)$ has fixed switching times at $s = 1, \ldots, N$, and its height represents the duration between consecutive harvest times in the original time horizon. For $s \in [k-1,\ k]$, it follows from equations (2.12) and (2.15) that

$$t(s) = \int_0^s u(\eta) d\eta = \sum_{j=1}^{k-1} \theta_j + \theta_k (s - k + 1) \tag{2.16a}$$

and thus

$$t(k) = \sum_{j=1}^{k} \theta_j = \sum_{j=1}^{k} (\tau_j - \tau_{j-1}). \tag{2.16b}$$

In particular, $t(N) = T$, as required by equation (2.12b).

Applying the transformation defined by (2.12), the dynamics in Problem 2A become

$$\dot{\widetilde{n}}(s) = -m\widetilde{n}(s)u(s), \qquad \widetilde{n}(0) = n_0, \tag{2.17}$$

$$\dot{\widetilde{w}}(s) = \big(a - b\widetilde{w}(s)\widetilde{n}(s)\big)u(s), \qquad \widetilde{w}(0) = w_0, \tag{2.18}$$

where $\widetilde{n}(s) = n(t(s)),\ \widetilde{w}(s) = w(t(s))$, and $t(s)$ is the solution of the differential equation (2.12a). In the new time horizon, the state jumps occur at the fixed times $s = 1, \ldots, N-1$. Thus, the jump conditions (2.5) and (2.6) become

$$\widetilde{n}(j^+) - \widetilde{n}(j^-) = -\nu_j \widetilde{n}(j^-), \tag{2.19}$$

$$\widetilde{w}(j^+) - \widetilde{w}(j^-) = 0. \tag{2.20}$$

Problem 2A is thus transformed into the following problem: Choose the control $u(s)$ and the parameters $\nu_j,\ j = 1, \ldots, N$ to maximize the transformed revenue function

$$\widetilde{J} = \sum_{j=1}^{N} \Big(10^{-3} \{p\big(\widetilde{w}(j)\big) - c_j\} \widetilde{w}(j) \widetilde{n}(j^-) \nu_j - h \Big) \tag{2.21}$$

subject to the dynamics (2.12), (2.17)-(2.18), the jump conditions (2.19)-(2.20) and the following bounds:

$$0 \leq \nu_j \leq 1, \quad j = 1, \ldots, N, \tag{2.22a}$$

$$0 \leq u(s) \leq T, \quad s \in [0, N]. \tag{2.22b}$$

We refer to this problem as Problem 2B. The control $u(s)$ in Problem 2B governs the harvesting times in the original time horizon.

Problems 2A and 2B are mathematically equivalent. The variable jump times in Problem 2A have been replaced by fixed times in Problem 2B. Although this makes the problem more amenable to solution via standard optimal control software packages such as MISER 3.3, the objective function is still a discontinuous function of the state. Thus, Problem 2B cannot be solved directly by MISER 3.3, which requires the objective to be smooth. In the next subsection, we overcome this difficulty by introducing new binary variables (adopting the transformation strategy described in [55, 131]) in addition to new linear and quadratic constraints.

2.3.2 Smoothing Transformation

Let z_{ij}, $i = 1, \ldots, L$, $j = 1, \ldots, N$, be new binary variables defined as follows:

$$z_{ij} = \begin{cases} 1, & \text{if } \beta_{i-1} \leq \widetilde{w}(j) < \beta_i, \\ 0, & \text{otherwise.} \end{cases}$$

With this definition, we can write $p\big(\widetilde{w}(j)\big)$ as:

$$p\big(\widetilde{w}(j)\big) = \sum_{i=1}^{L} z_{ij}\alpha_i.$$

Thus, the revenue function may be written as:

$$\widehat{J}(\mathbf{z}, \boldsymbol{\nu}, u) = \sum_{j=1}^{N} \left(10^{-3} \left\{ \sum_{i=1}^{L} z_{ij}\alpha_i - c_j \right\} \widetilde{w}(j)\widetilde{n}(j^-)\nu_j - h \right), \tag{2.23}$$

where

$$\begin{aligned} \boldsymbol{\nu} &= [\nu_1, \ldots, \nu_N]^\top \in \mathbb{R}^N \\ \mathbf{z}_j &= [z_{1j}, z_{2j}, \ldots, z_{Lj}]^\top \in \mathbb{R}^L \\ \mathbf{z} &= [(\mathbf{z}_1)^\top, (\mathbf{z}_2)^\top, \ldots, (\mathbf{z}_N)^\top]^\top \in \mathbb{R}^{LN} \end{aligned}$$

Although we have defined each z_{ij} to be a binary variable, MISER 3.3 only allows us to define variables in a continuous domain. Hence, to ensure that each z_{ij} is a binary variable, we impose the following constraints:

$$H_j(\mathbf{z}) = \sum_{i=1}^{L} z_{ij} - 1 = 0, \qquad j = 1, \ldots, N, \tag{2.24a}$$

$$g_{ij}(\mathbf{z}) = z_{ij}(1 - z_{ij}) \leq 0, \qquad i = 1, \ldots, L, \qquad j = 1, \ldots, N, \tag{2.24b}$$

$$0 \leq z_{ij} \leq 1, \qquad i = 1, \ldots, L, \qquad j = 1, \ldots, N. \tag{2.24c}$$

It is clear that (2.24) ensures $z_{ij} \in \{0,1\}$. However, (2.24) alone is not sufficient to ensure that z_{ij} is consistent with the definition given at the beginning of this section. Therefore, we impose the additional constraints given below:

$$G_{ij}(\mathbf{z}) = z_{ij}\big(\beta_{i-1} - \widetilde{w}(j)\big)\big(\beta_i - \widetilde{w}(j)\big) \leq 0, \quad i = 1,\ldots,L, \quad j = 1,\ldots,N. \qquad (2.25)$$

We now prove two important results.

Lemma 2.1. *Suppose that z_{ij}, $i = 1,\ldots,L$, $j = 1,\ldots,N$, satisfy constraints (2.24) and (2.25). For any $i \in \{1,\ldots,L\}$ and any $j \in \{1,\ldots,N\}$, if $z_{ij} = 1$, then $\beta_{i-1} \leq \widetilde{w}(j) \leq \beta_i$.*

Proof. Suppose $z_{ij} = 1$. Then inequality (2.25) reduces to

$$\big(\beta_{i-1} - \widetilde{w}(j)\big)\big(\beta_i - \widetilde{w}(j)\big) \leq 0.$$

Since $\beta_{i-1} < \beta_i$ (recall (2.9b)), this inequality is only satisfied when we have

$$\beta_{i-1} \leq \widetilde{w}(j) \leq \beta_i.$$

□

Lemma 2.2. *Suppose that z_{ij}, $i = 1,\ldots,L$, $j = 1,\ldots,N$, satisfy constraints (2.24) and (2.25). For any $i \in \{1,\ldots,L\}$ and any $j \in \{1,\ldots,N\}$, if $\beta_{i-1} < \widetilde{w}(j) < \beta_i$, then $z_{ij} = 1$.*

Proof. Suppose $\beta_{i-1} < \widetilde{w}(j) < \beta_i$. Recall that inequalities (2.24b) and (2.24c) ensure that $z_{ij} \in \{0,1\}$. Suppose that $z_{ij} \neq 1$. Then we must have $z_{ij} = 0$. It thus follows from (2.24a) that there exists a $k \in \{1,\ldots,L\} \setminus \{i\}$ such that $z_{kj} = 1$. Then, by Lemma 1, we have $\beta_{k-1} \leq \widetilde{w}(j) \leq \beta_k$. Since $k \neq i$, and the weight intervals are disjoint (recall (2.9b)), this contradicts $\beta_{i-1} < \widetilde{w}(j) < \beta_i$.

Having arrived at a contradiction, it follows that $z_{ij} = 1$. □

Remark 2.1. Lemma 2 implies that if $\widetilde{w}(j)$ lies in the interior of the i^{th} weight range, then we must have $z_{ij} = 1$. However, the converse of this result is not true in general *(i.e. Lemma 2 is not the direct converse of Lemma 1).* According to Lemma 1 and (2.9b), if $\widetilde{w}(j) = \beta_i$, then the only two possibilities are $z_{(i+1),j} = 1$ and $z_{ij} = 1$. Since the shrimp price increases with the average weight of shrimp, if $\widetilde{w}(j) = \beta_i$ at an optimal solution, then $z_{(i+1),j} = 1$ and $z_{ij} = 0$. Thus, the optimization process will push $\widetilde{w}(j)$ into a higher weight range as the revenue is maximized. It follows that, at an optimal solution, z_{ij} is an indicator variable equal to 1 if $\widetilde{w}(j)$ is in the i^{th} weight range and equal to zero otherwise.

Our transformed problem can hence be described as follows: Choose the system parameters z_{ij}, the harvesting fractions ν_j and the control $u(s)$ to maximize (2.23) subject to:

- the dynamics (2.12) and (2.17)-(2.18);
- the jump conditions (2.19)-(2.20);
- the bounds (2.22);
- the constraints (2.24)-(2.25).

We refer to this problem as Problem 2C.

Although Problem 2C has a smooth objective function, the additional quadratic constraints imposed on the system (see inequalities (2.24) and (2.25)) give rise to a disconnected region. Hence, standard optimization algorithms will struggle with these constraints. Indeed, when using MISER 3.3 to solve Problem 2C directly, we encountered a large number of numerical issues. This is expected, as MISER 3.3 assumes that the optimization problem has a continuous feasible region, an assumption that is violated in Problem 2C.

In the next section, we apply the exact penalty method proposed in [55] to transform Problem 2C into an unconstrained problem, which can then be solved readily by MISER 3.3.

2.4 An Exact Penalty Method

The exact penalty approach involves creating a pseudo-objective function by adding terms based on the constraints to the objective. Problem 2C, a constrained optimization problem, is subsequently transformed into an approximate unconstrained problem that can be readily solved using MISER 3.3.

The *constraint violation* is defined by:

$$\Delta(\mathbf{z},u) = \sum_{j=1}^{N} \big(H_j(\mathbf{z})\big)^2 + \sum_{i=1}^{L}\sum_{j=1}^{N} \big(\max\{0, G_{ij}(\mathbf{z})\}\big)^2 + \sum_{i=1}^{L}\sum_{j=1}^{N} \big(\max\{0, g_{ij}(\mathbf{z})\}\big)^2 + (t(N)-T)^2.$$

Note that $\Delta(\mathbf{z},u)=0$ if and only if the constraints in Problem 2C are satisfied.

Using the strategy introduced in [130, 131], an exact penalty function $\widehat{J}_\sigma(\mathbf{z},\boldsymbol{\nu},u,\epsilon)$ is constructed as follows:

$$\widehat{J}_\sigma(\mathbf{z},\boldsymbol{\nu},u,\epsilon) = \begin{cases} -\widehat{J}(\mathbf{z},\boldsymbol{\nu},u), & \text{if } \epsilon = 0,\ \Delta(\mathbf{z},u)=0, \\ -\widehat{J}(\mathbf{z},\boldsymbol{\nu},u) + \epsilon^{-\lambda}\Delta(\mathbf{z},u) + \sigma\epsilon^{\gamma}, & \text{if } \epsilon > 0, \\ +\infty, & \text{otherwise,} \end{cases}$$

where

- ϵ is a new decision variable;
- $\sigma > 0$ is the penalty parameter;
- λ and γ are positive constants satisfying $1 \leq \gamma \leq \lambda$.

The new decision variable ϵ is subject to the following bounds:

$$0 \leq \epsilon \leq \widetilde{\epsilon}, \tag{2.26}$$

where $\widetilde{\epsilon} > 0$ is a small positive number.

Our unconstrained penalty problem, to be referred to as Problem 2D, is defined as follows: Choose the system parameters z_{ij}, the harvesting fractions ν_j, the new decision variable ϵ and the control $u(s)$ to minimize $\widehat{J}_\sigma(\mathbf{z}, \boldsymbol{\nu}, u, \epsilon)$ subject to:

- the dynamics (2.12) and (2.17)-(2.18);
- the jump conditions (2.19)-(2.20);
- the bounds (2.22) and (2.26).

Note that when the penalty parameter σ is large, the term $\sigma\epsilon^\gamma$ in $\widehat{J}_\sigma$ forces ϵ to be small, thus causing the second term $\epsilon^{-\lambda}\Delta(\mathbf{z}, u)$ to severely penalize any constraint violations. It can be shown that $\widehat{J}_\sigma$ is an exact penalty function in the sense that when the penalty parameter σ is sufficiently large, any local solution of the approximate unconstrained problem (i.e. Problem 2D) is also a local solution of Problem 2C [55].

Problem 2D can be solved as a non-linear programming problem using the MISER 3.3 software. MISER automatically calculates the objective function gradients by integrating a costate system backwards in time; for more details see [14, 34, 54, 66].

In summary, the original non-smooth optimal control problem undergoes a series of transformations to overcome the challenges that would exist in realistic problems of this nature. This approach is summarized below.

- Our original problem included a complex revenue function that is a discontinuous function of the state variables. The decision variables need to be chosen optimally to maximize this revenue function.
- We use the time-scaling transformation (described in subsection 2.3.1) to map the variable jump times to fixed times in a new time horizon, as standard optimal control algorithms can only deal with fixed jump times.
- We then use a smoothing transformation involving binary variables and quadratic constraints (described in subsection 2.3.2) to overcome the challenge posed by the discontinuous objective function.
- Since standard optimal control software such as MISER 3.3 struggle with these quadratic constraints, we apply the exact penalty approach to transform the problem into an unconstrained problem. We thus arrive at a smooth impulsive optimal control problem with fixed jump times and only bound constraints. Such problems can be solved effectively using MISER 3.3, which solves optimal control problems using non-linear programming techniques.

i	β_{i-1}	β_i	α_i
1	0	5	$2
2	5	10	$4
3	10	15	$6
4	15	20	$8
5	20	25	$12

Table 2.1: Price function parameters

In the next section, we demonstrate the efficiency of the proposed approach with a numerical example.

2.5 Numerical Results

We consider the shrimp farming model described in Section 2 with the following parameters:

- $N = 4$ (3 partial harvests and 1 final harvest);
- $L = 5$ (price function is based on 5 different weight ranges for the shrimp);
- $T = 13.2$;
- $c_j = 0$ for $j = 1, 2, 3, 4$ (no variable harvesting costs);
- $h = 50$;
- $n_0 = 40,000$ and $w_0 = 1$;
- $a = 3.5$ and $b = 10^{-5}$;
- α_i, $i = 1, \ldots, 5$ and β_i, $i = 0, \ldots, 5$ are given in Table 2.1.

Recall from Section 2.4 that the parameters λ and γ in the exact penalty function must satisfy the condition $1 \leq \gamma \leq \lambda$. Numerical testing reveals that the choice of λ and γ can significantly affect the results. Our best results were obtained using either $\lambda = 4.01$ and $\gamma = 3.55$ or $\lambda = 5.01$ and $\gamma = 3.55$. These results are presented below.

When running MISER 3.3, the initial values for ϵ and σ were 5.0×10^{-1} and 1.0×10^4 respectively. The penalty parameter σ was increased by a multiple of 10 for each subsequent MISER run. As expected this caused ϵ to decrease in value. Tables 2 and 3 show the penalty function value and the optimal value of ϵ (denoted by ϵ^*) for each run.

Note that the results in Tables 2.2 and 2.3 show a clear convergence of the objective function as σ is increased.

The optimal solution corresponding to the last line in Table 2.2 is:

$$\nu_1 = 6.8488 \times 10^{-1}, \qquad \nu_2 = 6.7209 \times 10^{-1}, \qquad \nu_3 = 0.0000, \qquad \nu_4 = 1.0000,$$

σ	ϵ^*	Penalty Function Value	Constraint Violation
10^4	4.0661×10^{-1}	5.601975×10^3	0.0000
10^5	2.6733×10^{-1}	4.180581×10^3	0.0000
10^6	2.8292×10^{-2}	3.107395×10^3	0.0000
10^7	5.0437×10^{-4}	3.110452×10^3	0.0000
10^8	7.5006×10^{-4}	3.110451×10^3	0.0000

Table 2.2: Numerical convergence using $\lambda = 4.01$ and $\gamma = 3.55$

σ	ϵ^*	Penalty Function Value	Constraint Violation
10^4	4.3980×10^{-1}	5.330159×10^3	0.0000
10^5	2.9917×10^{-1}	4.330777×10^3	0.0000
10^6	1.6207×10^{-2}	3.106807×10^3	0.0000
10^7	8.2666×10^{-3}	3.107617×10^3	0.0000
10^8	8.2661×10^{-3}	3.103999×10^3	0.0000

Table 2.3: Numerical convergence using $\lambda = 5.01$ and $\gamma = 3.55$

$$\tau_1 = 8.47906, \qquad \tau_2 = 13.2, \qquad \tau_3 = 13.2, \qquad \tau_4 = 13.2,$$
$$w(\tau_1) = 10.0, \qquad w(\tau_2) = 20.0, \qquad w(\tau_3) = 20.0, \qquad w(\tau_4) = 20.0.$$

The optimal solution corresponding to the last line in Table 2.3 is:

$$\nu_1 = 6.4803 \times 10^{-1}, \qquad \nu_2 = 5.4889 \times 10^{-2}, \qquad \nu_3 = 8.6884 \times 10^{-2}, \qquad \nu_4 = 1.0000,$$
$$\tau_1 = 8.47906, \qquad \tau_2 = 8.89985, \qquad \tau_3 = 10.71427, \qquad \tau_4 = 13.2,$$
$$w(\tau_1) = 10.0, \qquad w(\tau_2) = 10.9935, \qquad w(\tau_3) = 15.0, \qquad w(\tau_4) = 20.0.$$

The optimal state variables corresponding to the solutions in Tables 2.2 and 2.3 are shown in Figures 2.1 and 2.2.

The results obtained in this section cannot be directly compared to the numerical results obtained in [53, 133]. This is because the price function used here is a weight dependent piecewise constant function (recall (2.9a)) and is not a fixed constant as in the numerical examples of [53, 133]. The method presented in this chapter is more realistic from a commercial point of view, as the shrimp price is expected to be heavily dependent on the weight of the shrimp.

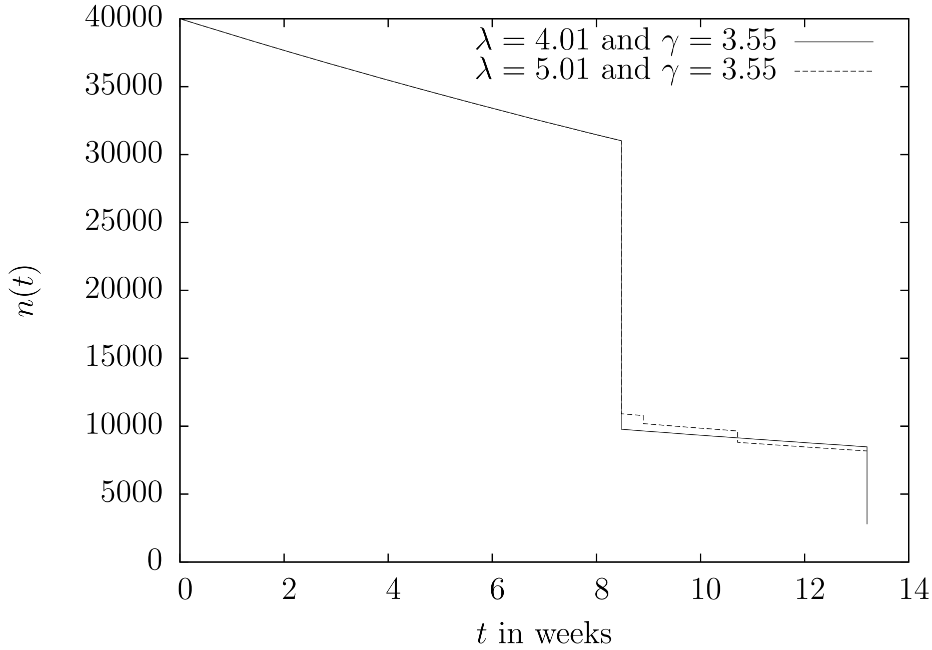

Figure 2.1: Number of shrimps

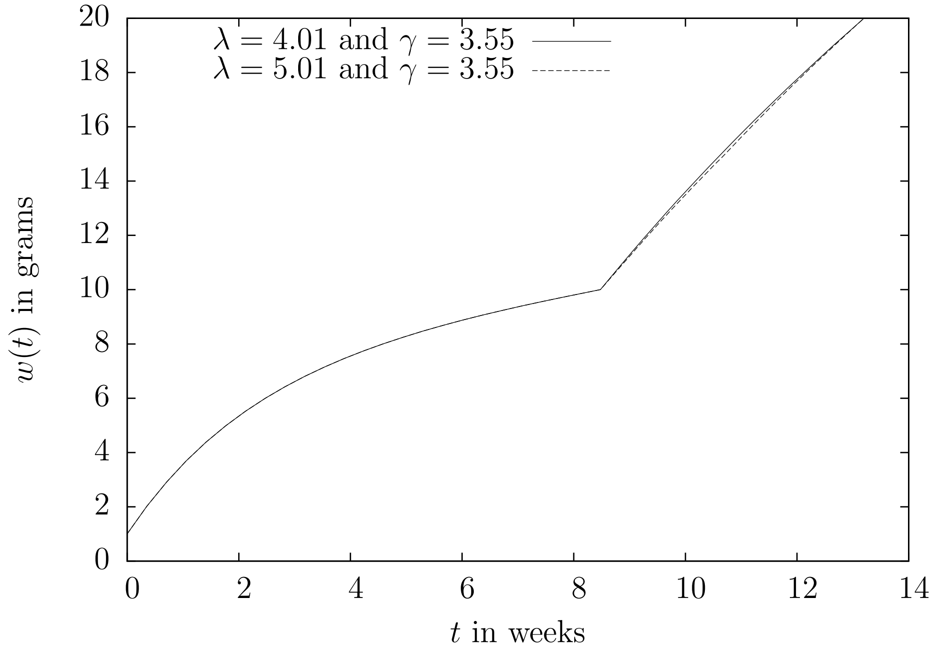

Figure 2.2: Average weight of shrimp

2.6 Concluding Remarks

We have developed an efficient computational algorithm for solving a class of optimization problems containing a discontinuous (and therefore non-differentiable) objective function subject to a dynamic system involving jump conditions at variable time points. The technique was successfully tested on a realistic shrimp farming problem. We note that that this algorithm could be adapted to other classes of problems to maximize or minimize non-smooth objective functions subject to various forms of constraints and continuous time dynamics. Our results illustrate that we can obtain clear convergence of the objective function while optimally determining the partial harvesting fractions as well the corresponding partial harvest times, as shown by the numerical results.

CHAPTER 3

Impulsive Switched Systems with Minimum Subsystem Durations

3.1 Introduction

An impulsive switched system is a dynamic system whose state and state dynamics undergo instantaneous changes at certain times in the time horizon. These times, called jump times or switching times, are usually decision variables to be chosen optimally by the system operator. In practical applications, the goal is to optimize the impulsive switched system by manipulating the jump times, as well as other control parameters influencing the state dynamics and/or state jumps, to achieve optimal system performance. Impulsive switched systems arise in many practical applications including switching DC–DC power converters [31, 67], shrimp harvesting [133] and communication security [40].

Consider an impulsive switched system consisting of N subsystems operating in succession. Let τ_i denote the i^{th} switching time. Then $\tau_i, i = 1, \ldots, N$, satisfy the following ordering constraints:

$$0 = \tau_0 \leq \tau_1 \leq \tau_2 \leq \cdots \leq \tau_N. \tag{3.1}$$

In the literature on the optimal control of impulsive switched systems, the inequalities in (3.1) are usually the only constraints imposed on the switching times (see, for example, references [37, 40, 60, 80, 119, 122, 124]). However, the constraints in (3.1) do not preclude very short operating durations for some subsystems. In reality, it is physically impossible to switch too frequently, and hence there is often a minimum duration for which a subsystem must be active. A first attempt at addressing this issue was made in [93], where a penalty on short subsystem durations was added to the objective function. However, this is only a heuristic approach and there is no guarantee that it yields optimal (or even feasible) results for a general class of impulsive switched systems.

References [60] consider an optimal control problems governed by an impulsive system in which the state variables (but not the state dynamics) experience abrupt jumps at a finite number of jump times. The magnitudes of the state jumps are controlled through a set of system parameters. To solve these problems in [60], the authors developed a transformation procedure that involves introducing a new time variable and mapping the jump times (which are decision variables to be chosen optimally) into fixed points in a new time horizon. The resulting problems in [60] and [119] are solved using the optimal control software MISER 3.2 and MISER 3.3 [34] respectively. The technique designed in [60] usually performs reasonably well in practice, although it is only capable of finding locally optimal solutions. In [122], this limitation was overcome by combining the gradient-based optimization technique in [60] with a filled function method for finding globally optimal solutions. The authors in [119] use a two-stage numerical technique to solve the impulsive optimal control problem. However, they use the control parameterization in conjunction with the time-scaling and then use MISER3.3 to solve the first part of their problem.

Although the methods in [60, 122] usually work well, they do not have the capability to preclude extremely short operating durations. In fact, the numerical example in [60] has solutions in which some of the subsystem durations become very short and disappear altogether when the number of allowed state jumps is increased. Due to the formulation of the problem class in [60, 122], when a subsystem duration becomes zero during the optimization process, the two corresponding state jumps still occur. But it would be more appropriate in this situation if one of these state jumps is deleted, so that only one jump occurs at the common jump time. The methods in [60, 122] do not have this capability: they instead impose "multi-jumps" when two or more switching times coincide. This challenge was overcome in [66] by combining the time-scaling transformation from [60,122] with a penalty technique. The numerical results in [66] demonstrate that subsystems can be deleted in the optimal solution without the occurrence of multiple state jumps.

More recently in [53], a new algorithm for solving optimal control problems governed by impulsive switched systems was developed. This algorithm is based on a new way of applying the time-scaling transformation, as well as a new method for computing the cost function's gradient via the solution of an auxiliary dynamic system. The disadvantage with the algorithm in [53] is that *every* potential subsystem is assumed to operate for a positive duration. Hence, unlike in [66], there is no scope to remove non-optimal subsystems.

In this chapter, we assume that each subsystem either does not operate (duration of zero) or operates for a minimum non-negligible amount of time (duration is no less than a specified minimum positive number). This requires that adjacent switching times satisfy the constraint $\tau_i - \tau_{i-1} \in \{0\} \cup [\epsilon_i, \infty)$, where τ_i is as defined in (3.1) and $\epsilon_i > 0$ is the minimum duration of the i^{th} subsystem. Unlike (3.1), this constraint defines a disconnected region for the subsystem durations, thus causing problems for standard optimization algorithms such as sequential quadratic programming [70, 77]. In this chapter, we introduce a novel exact penalty function to overcome this challenge.

The class of problems considered in this chapter is more general than the class of problems considered in [66]. In particular, we allow canonical state constraints, whereas no state constraints are considered in [66]. Furthermore, we develop a transformation procedure that transforms the problem with disconnected region into an *equivalent* standard dynamic optimization problem. In contrast, the approach in [66] is based on an approximation scheme and does not maintain equivalence.

This chapter is organized as follows. In Section 3.2, we define the optimal control problem under consideration. Then, in Subsection 3.3.1, we use the time-scaling technique [53] to map the variable jump times to fixed integers in a new time horizon. Following this, we implement another transformation by introducing new binary variables in Subsection 3.3.2. However, the constraints on the binary variables define a disconnected region, which poses a challenge for standard optimal control software such as MISER 3.3. Therefore, we adopt an exact penalty approach in Subsection 3.3.3 to transform the problem into a sequence of unconstrained problems with fixed jump times. Each of these unconstrained problems is a smooth impulsive optimal control problem that can be solved effectively by MISER 3.3. Our numerical example in Section 3.4 provides evidence that the proposed method is effective and efficient.

3.2 Problem Formulation

Consider the following impulsive switched system:

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}^i(\boldsymbol{x}(t), \boldsymbol{\zeta}), \qquad t \in (\tau_{i-1}, \tau_i), \qquad i = 1, \ldots, N, \tag{3.2}$$

with jump conditions

$$\boldsymbol{x}(\tau_i^+) = \begin{cases} \boldsymbol{x}^0, & \text{if } i = 0, \quad (3.3a) \\ \boldsymbol{x}(\tau_i^-) + \boldsymbol{\phi}^i(\boldsymbol{x}(\tau_i^-), \boldsymbol{\zeta}), & \text{if } i \in \{1, \ldots, N\} \text{ and } \tau_i - \tau_{i-1} \geq \epsilon_i, \quad (3.3b) \end{cases}$$

where

- $\boldsymbol{x}(t) \in \mathbb{R}^n$ is the state at time t;
- $\boldsymbol{\zeta} \in \mathbb{R}^r$ is a vector of control parameters;
- $\boldsymbol{x}^0 \in \mathbb{R}^n$ is a given initial state;
- τ_i, $i = 1, \ldots, N-1$, are jump or switching times, with $\tau_0 = 0$ and $\tau_N = T$;
- $T > 0$ is a given terminal time;
- $\epsilon_i > 0$ is the given minimum duration of the i^{th} subsystem;
- $\boldsymbol{f}^i : \mathbb{R}^n \times \mathbb{R}^r \to \mathbb{R}^n$ and $\boldsymbol{\phi}^i : \mathbb{R}^n \times \mathbb{R}^r \to \mathbb{R}^n$, $i = 1, \ldots, N$, are given functions assumed to be continuously differentiable.

Note that the positive and negative superscripts in (3.3b) mean $\boldsymbol{x}(\tau^\pm) = \lim_{t \to \tau^\pm} \boldsymbol{x}(t)$. Note also that if $\tau_i = T$ in (3.3b), then $\tau_i^+ = T$ by convention.

System (3.2)-(3.3) consists of N potential subsystems operating in succession. The dynamic behaviour of the i^{th} subsystem is governed by the function $\boldsymbol{f}^i$. At the switching times τ_i, $i = 1, \ldots, N-1$, the system changes from one subsystem to another, and this causes an instantaneous jump in the system state according to (3.3b).

Note that if τ_{i-1} and τ_i coincide (i.e. $\tau_{i-1} = \tau_i$), then the i^{th} subsystem does not operate. This is allowed, as running every potential subsystem may not be optimal. The condition $\tau_i - \tau_{i-1} \geq \epsilon_i$ in (3.3b) ensures that the state jump occurring at the end of a certain subsystem is only imposed if that particular subsystem runs for a non-negligible amount of time. This is different to the conventional impulsive switched systems considered in the literature (see, for example, [60, 119, 122]), which impose state jumps even for subsystems that do not operate. For example, if $\tau_1 = \tau_2$ in a conventional impulsive switched system, then a "double jump" – one for subsystem 1 and another for subsystem 2 – will be imposed at $t = \tau_1 = \tau_2$, even though subsystem 2 does not actually operate. This is usually not an accurate reflection of the real system under consideration.

The control parameter vector $\boldsymbol{\zeta} = [\zeta_1, \ldots, \zeta_r]^\top \in \mathbb{R}^r$ is subject to the following bound constraints:

$$a_j \leq \zeta_j \leq b_j, \qquad j = 1, \ldots, r, \tag{3.4}$$

where a_j and b_j are given constants such that $a_j < b_j$. Let $\mathcal{Z}$ denote the set of all $\boldsymbol{\zeta} \in \mathbb{R}^r$ satisfying (3.4).

We also have the following constraints on the switching times:

$$\tau_i - \tau_{i-1} \in \{0\} \cup [\epsilon_i, T], \qquad i = 1, \ldots, N. \tag{3.5}$$

Thus, either subsystem i runs for a duration of at least ϵ_i time units, or it does not run at all. This means that it is possible to "delete" certain subsystems if it is optimal to do so, which may be necessary when the optimal number of switches is unknown. Clearly, constraint (3.5) is more complex than constraint (3.1) given earlier, which is a simple ordering constraint on the switching times. In particular, (3.1) is convex, but (3.5) is non-convex. In fact, imposing (3.5) leads to a disconnected region for the subsystem durations.

Let Γ denote the set of all $\boldsymbol{\tau} = [\tau_1, \ldots, \tau_{N-1}]^\top \in \mathbb{R}^{N-1}$ satisfying constraint (3.5). Furthermore, let $\boldsymbol{x}(\cdot|\boldsymbol{\tau}, \boldsymbol{\zeta})$ denote the right-continuous solution of (3.2)-(3.3) corresponding to the given pair $(\boldsymbol{\tau}, \boldsymbol{\zeta}) \in \Gamma \times \mathcal{Z}$.

We suppose that the system is subject to the following canonical constraints:

$$G_j(\boldsymbol{\tau}, \boldsymbol{\zeta}) = \sum_{i=1}^{N} \widehat{\Phi}_{j,i}(\tau_i - \tau_{i-1}, \boldsymbol{x}(\tau_i|\boldsymbol{\tau}, \boldsymbol{\zeta}), \boldsymbol{\zeta}) \begin{cases} = 0, & j = 1, \ldots, q_e, \quad (3.6a) \\ \geq 0, & j = q_e + 1, \ldots, q, \quad (3.6b) \end{cases}$$

where

$$\widehat{\Phi}_{j,i}(\tau_i - \tau_{i-1}, \boldsymbol{x}(\tau_i|\boldsymbol{\tau}, \boldsymbol{\zeta}), \boldsymbol{\zeta}) = \begin{cases} \Phi_{j,i}(\boldsymbol{x}(\tau_i|\boldsymbol{\tau}, \boldsymbol{\zeta}), \boldsymbol{\zeta}), & \text{if } \tau_i - \tau_{i-1} \geq \epsilon_i, \quad (3.7a) \\ 0, & \text{if } \tau_i - \tau_{i-1} = 0, \quad (3.7b) \end{cases}$$

and $\Phi_{j,i} : \mathbb{R}^n \times \mathbb{R}^r \to \mathbb{R}$ are given functions assumed to be continuously differentiable with respect to each of their arguments. Here, the definition of $\widehat{\Phi}_{j,i}$ ensures that each distinct switch contributes exactly once to the j^{th} constraint. More specifically, if $\tau_i = \tau_{i-1}$, then the i^{th} term in the summation on the right-hand side of (3.6) is set equal to zero to avoid "doubling up" when multiple switching times coincide. For example, if $\tau_1 = \tau_2 > 0$, then $\widehat{\Phi}_{j,1} = \Phi_{j,1}$, but $\widehat{\Phi}_{j,2} = 0$; that is, there is no contribution from the second subsystem to the j^{th} constraint because the second subsystem does not actually operate.

The system cost function defined below is assumed to take the same canonical form as the constraints in (3.6):

$$G_0(\boldsymbol{\tau},\boldsymbol{\zeta}) = \sum_{i=1}^{N} \widehat{\Phi}_{0,i}(\tau_i - \tau_{i-1}, \boldsymbol{x}(\tau_i|\boldsymbol{\tau},\boldsymbol{\zeta}), \boldsymbol{\zeta}), \tag{3.8}$$

where $\widehat{\Phi}_{0,i}$ is defined in a similar way to $\widehat{\Phi}_{j,i}$ in (3.7).

Note that the canonical functions in (3.7) and (3.8) are discontinuous functions of the subsystem durations. This formulation is in contrast to the usual definition of canonical functions in optimal control [34,52], and has, to the best of our knowledge, not previously been considered in the literature.

Our problem is to choose $(\boldsymbol{\tau},\boldsymbol{\zeta}) \in \Gamma \times \mathcal{Z}$ to minimize the cost function given by (3.8) subject to the governing impulsive switched system given by (3.2) and (3.3), the bounds (3.4), the non-convex constraints (3.5) and the canonical constraints (3.6). We refer to this problem as Problem 3A. We assume throughout this chapter that a solution exists for Problem 3A.

3.3 Solution Method

3.3.1 Time-Scaling Transformation

It is well known in computational optimal control that standard numerical optimization algorithms are not effective at optimizing variable switching times [37, 54]. Thus, in this section, we will apply the time-scaling transformation described in [52, 53, 60] to map the variable switching times to fixed times in a new time horizon. This yields an equivalent problem in which the variable switching times are replaced by conventional decision parameters.

We first introduce a new time variable $s \in [0, N]$ and relate s to t through the following differential equation:

$$\dot{t}(s) = \sum_{i=1}^{N} \theta_i \chi_{[i-1,i)}(s), \qquad t(0) = 0, \tag{3.9a}$$

and

$$t(N) = T, \tag{3.9b}$$

where $\theta_i = \tau_i - \tau_{i-1}$ is the duration of the i^{th} subsystem and, for a given interval I, $\chi_I(s)$ is the corresponding indicator function defined by

$$\chi_I(s) = \begin{cases} 1, & \text{if } s \in I, \\ 0, & \text{otherwise.} \end{cases}$$

Constraints (3.5) can be equivalently expressed in terms of the following inequalities:

$$(\tau_i - \tau_{i-1})(\epsilon_i - \tau_i + \tau_{i-1}) \leq 0, \qquad i = 1, \ldots, N, \tag{3.10}$$

and

$$\tau_i \geq \tau_{i-1}, \qquad i = 1, \ldots, N. \tag{3.11}$$

Based on the definition of θ_i, constraints (3.10) and (3.11) become:

$$\theta_i(\epsilon_i - \theta_i) \leq 0, \qquad i = 1, \ldots, N, \tag{3.12a}$$

and

$$\theta_i \geq 0, \qquad i = 1, \ldots, N. \tag{3.12b}$$

For $s \in [i-1, i]$, it follows from (3.9) that

$$t(s) = \int_0^s \dot{t}(\eta)d\eta = \sum_{k=1}^{i-1} \theta_k + \theta_i(s - i + 1). \tag{3.13}$$

Thus, for each $i = 0, \ldots, N$,

$$t(i) = \sum_{k=1}^{i} \theta_k = \sum_{k=1}^{i} (\tau_k - \tau_{k-1}) = \tau_i. \tag{3.14}$$

In particular, $t(N) = \tau_N = T$, as required by equation (3.9b).

Let $\widetilde{\boldsymbol{x}}(s) = \boldsymbol{x}(t(s))$, where $t(s)$ is the solution of the differential equation (3.9a). Since $\theta_i \geq 0$, it is clear that t is a non-decreasing function of s. Thus, if $s \in (i-1, i)$, then

$$t(s) \in [t(i-1), t(i)] = [\theta_1 + \cdots + \theta_{i-1}, \theta_1 + \cdots + \theta_i]. \tag{3.15a}$$

In fact, if $\theta_i > 0$, then

$$t(s) \in (\theta_1 + \cdots + \theta_{i-1}, \theta_1 + \cdots + \theta_i), \qquad s \in (i-1, i), \tag{3.15b}$$

and thus

$$\frac{d}{ds}\{\widetilde{\boldsymbol{x}}(s)\} = \frac{d}{ds}\{\boldsymbol{x}(t(s))\} = \dot{t}(s)\dot{\boldsymbol{x}}(t(s)) = \theta_i \boldsymbol{f}^i(\widetilde{\boldsymbol{x}}(s), \boldsymbol{\zeta}), \qquad s \in (i-1, i). \tag{3.16a}$$

On the other hand, if $\theta_i = 0$, then $t(s) = \theta_1 + \cdots + \theta_{i-1}$ for all $s \in (i-1, i)$, and thus

$$\frac{d}{ds}\{\widetilde{\boldsymbol{x}}(s)\} = \frac{d}{ds}\{\boldsymbol{x}(t(s))\} = \frac{d}{ds}\{\boldsymbol{x}(\theta_1 + \cdots + \theta_{i-1})\} = \boldsymbol{0}, \qquad s \in (i-1, i). \tag{3.16b}$$

Combining (3.16a) and (3.16b) shows that, under the time-scaling transformation, the system dynamics in Problem 3A become

$$\dot{\widetilde{\boldsymbol{x}}}(s) = \theta_i \boldsymbol{f}^i(\widetilde{\boldsymbol{x}}(s), \boldsymbol{\zeta}), \qquad s \in (i-1, i), \qquad i = 1, \ldots, N. \tag{3.17}$$

Since the time-scaling transformation maps $s = i$ to $t = \tau_i$ (see (3.14)), the state jumps in the new time horizon occur at the fixed times $s = 1, \ldots, N$. If $\theta_i > 0$, then $\theta_i \geq \epsilon_i$, and thus $t(i) - t(i-1) = \theta_i \geq \epsilon_i$. It therefore follows from (3.3b) that

$$\widetilde{\boldsymbol{x}}(i^+) = \widetilde{\boldsymbol{x}}(i^-) + \boldsymbol{\phi}^i(\widetilde{\boldsymbol{x}}(i^-), \boldsymbol{\zeta}).$$

If, on the other hand, $\theta_i = 0$, then $t(i) - t(i-1) = \theta_i = 0$. Hence,

$$\widetilde{\boldsymbol{x}}(i^+) = \widetilde{\boldsymbol{x}}(i^-).$$

Consequently, for each $i = 1, \ldots, N$, the jump conditions (3.3) become

$$\widetilde{\boldsymbol{x}}(i^+) = \begin{cases} \boldsymbol{x}^0, & \text{if } i = 0, \quad (3.18a) \\ \widetilde{\boldsymbol{x}}(i^-) + \boldsymbol{\phi}^i(\widetilde{\boldsymbol{x}}(i^-), \boldsymbol{\zeta}), & \text{if } i \in \{1, \ldots, N\} \text{ and } \theta_i \geq \epsilon_i, \quad (3.18b) \\ \widetilde{\boldsymbol{x}}(i^-), & \text{if } i \in \{1, \ldots, N\} \text{ and } \theta_i = 0. \quad (3.18c) \end{cases}$$

Note that by convention, $i^+ = N$ when $i = N$ in (3.18). Let $\widetilde{\boldsymbol{x}}(\cdot|\boldsymbol{\theta}, \boldsymbol{\zeta})$ denote the unique right-continuous solution of (3.17) and (3.18) corresponding to $\boldsymbol{\theta} = [\theta_1, \ldots, \theta_N]^\top \in \mathbb{R}^N$ and $\boldsymbol{\zeta} = [\zeta_1, \ldots, \zeta_r]^\top \in \mathbb{R}^r$.

Under the time-scaling transformation, the canonical constraints given by (3.6) become

$$\widetilde{G}_j(\boldsymbol{\theta}, \boldsymbol{\zeta}) = \sum_{i=1}^{N} \widehat{\Phi}_{j,i}(\theta_i, \widetilde{\boldsymbol{x}}(i|\boldsymbol{\theta}, \boldsymbol{\zeta}), \boldsymbol{\zeta}) \begin{cases} = 0, & j = 1, \ldots, q_e, \quad (3.19a) \\ \geq 0, & j = q_e + 1, \ldots, q, \quad (3.19b) \end{cases}$$

where $\widehat{\Phi}_{j,i}$ is defined by (3.7). Problem 3A is thus transformed into the following problem: Choose $(\boldsymbol{\theta}, \boldsymbol{\zeta}) \in \mathbb{R}^N \times \mathbb{R}^r$ to minimize the transformed cost function

$$\widetilde{G}_0(\boldsymbol{\theta}, \boldsymbol{\zeta}) = \sum_{i=1}^{N} \widehat{\Phi}_{0,i}(\theta_i, \widetilde{\boldsymbol{x}}(i|\boldsymbol{\theta}, \boldsymbol{\zeta}), \boldsymbol{\zeta}) \tag{3.20}$$

subject to the dynamics (3.9) and (3.17), the constraints (3.4), (3.12) and (3.19) and the jump conditions (3.18). We refer to this problem as Problem 3B.

Problems 3A and 3B are mathematically equivalent. Thus, a solution of Problem 3A can be used to generate a solution of Problem 3B, and vice versa. Note that the variable jump times in Problem 3A have been replaced by fixed jump times in Problem 3B. This makes Problem 3B much easier to deal with from a computational point of view, as variable jump times cause numerical difficulties, which are explained in [63]. However, Problem 3B still cannot be solved using conventional impulsive control techniques. This is because the jump conditions (3.18) and the canonical functions in (3.19) and (3.20) are expressed as discontinuous piecewise functions of θ_i. In fact, (3.18) can be written as follows:

$$\widetilde{\boldsymbol{x}}(i^+) = \begin{cases} \boldsymbol{x}^0, & \text{if } i = 0, \quad (3.21a) \\ \widetilde{\boldsymbol{x}}(i^-) + \chi_{[\epsilon_i, \infty)}(\theta_i)\boldsymbol{\phi}^i(\widetilde{\boldsymbol{x}}(i^-), \boldsymbol{\zeta}), & \text{if } i \in \{1, \ldots, N\}, \quad (3.21b) \end{cases}$$

where

$$\chi_{[\epsilon_i,\infty)}(\theta_i) = \begin{cases} 1, & \text{if } \theta_i \geq \epsilon_i, \\ 0, & \text{otherwise.} \end{cases}$$

Clearly, the presence of the indicator function $\chi_{[\epsilon_i,\infty)}$ causes the jump conditions (3.21) to be discontinuous. In [66], this difficulty was tackled by approximating the indicator function by a continuously differentiable function that depends on a smoothing parameter. However, to achieve sufficient accuracy, the value of the smoothing parameter must be reduced to the point where it leads to numerical difficulties.

In the next section, we propose a new approach that involves introducing binary decision variables to transform the jump conditions and the canonical functions into smooth forms. The advantage of this new approach is that it yields an equivalent problem, not just an approximation.

3.3.2 Transforming the Jump Conditions and Canonical Functions

Let v_i, $i = 1, \ldots, N$, be new binary decision variables defined as follows:

$$v_i = \begin{cases} 1, & \text{if } \theta_i \geq \epsilon_i, & (3.22\text{a}) \\ 0, & \text{if } \theta_i = 0. & (3.22\text{b}) \end{cases}$$

Using these binary variables, the jump conditions in (3.18) can be written in a more compact form as

$$\widetilde{\boldsymbol{x}}(i^+) = \begin{cases} \boldsymbol{x}^0, & \text{if } i = 0, & (3.23\text{a}) \\ \widetilde{\boldsymbol{x}}(i^-) + v_i \boldsymbol{\phi}^i(\widetilde{\boldsymbol{x}}(i^-), \boldsymbol{\zeta}), & \text{if } i \in \{1, \ldots, N\}. & (3.23\text{b}) \end{cases}$$

As in (3.18), we use the convention $N^+ = N$ here.

Let $\boldsymbol{v} = [v_1, v_2, \ldots, v_N]^\top$. Furthermore, let $\widetilde{\boldsymbol{x}}(\cdot|\boldsymbol{v}, \boldsymbol{\theta}, \boldsymbol{\zeta})$ denote the right-continuous solution of (3.17) and (3.23) corresponding to $(\boldsymbol{v}, \boldsymbol{\theta}, \boldsymbol{\zeta})$. Then the canonical constraints in (3.19) can be written as

$$\widetilde{G}_j(\boldsymbol{v}, \boldsymbol{\theta}, \boldsymbol{\zeta}) = \sum_{i=1}^{N} v_i \Phi_{j,i}(\widetilde{\boldsymbol{x}}(i|\boldsymbol{v}, \boldsymbol{\theta}, \boldsymbol{\zeta}), \boldsymbol{\zeta}) \begin{cases} = 0, & j = 1, \ldots, q_e, & (3.24\text{a}) \\ \geq 0, & j = q_e + 1, \ldots, q. & (3.24\text{b}) \end{cases}$$

Standard optimization algorithms such as interior-point methods and sequential quadratic programming (see [70, 77]) cannot handle binary variables in the form of (3.22). Therefore, to proceed, we drop the binary requirements and consider each v_i as a continuous optimization variable subject to the following constraints:

$$0 \leq v_i \leq 1, \quad i = 1, \ldots, N, \tag{3.25}$$

and

$$g_i(v_i) = v_i(1 - v_i) \leq 0, \quad i = 1, \ldots, N. \tag{3.26}$$

It is clear that (3.25) and (3.26) imply $v_i \in \{0, 1\}$. However, we also need to ensure that v_i is consistent with the definition given in (3.22). Thus, we impose the following additional constraints:

$$h_i(v_i, \theta_i) = (\epsilon_i - \theta_i)v_i \leq 0, \quad i = 1, \ldots, N, \tag{3.27}$$

and

$$H_i(v_i, \theta_i) = \theta_i(1 - v_i) \leq 0, \quad i = 1, \ldots, N. \tag{3.28}$$

We now prove that definition (3.22) is equivalent to (3.25)-(3.28).

Theorem 3.1. *Suppose that v_i, $i = 1, \ldots, N$, satisfy constraints* (3.25) *and* (3.26)*. Then for each $i \in \{1, \ldots, N\}$,* (3.22) *holds if only if* (3.27) *and* (3.28) *hold.*

Proof. First, suppose that (3.22) is satisfied. Then $v_i = 0$ implies $\theta_i = 0$ and $h_i(v_i, \theta_i) = 0$. Furthermore, $\theta_i = 0$ implies $H_i(v_i, \theta_i) = 0$.

Similarly, $v_i = 1$ implies $\theta_i \geq \epsilon_i$, $h_i(v_i, \theta_i) = \epsilon_i - \theta_i \leq 0$ and $H_i(v_i, \theta_i) = 0$. Therefore, (3.27) and (3.28) are satisfied.

Conversely, suppose inequalities (3.27) and (3.28) are satisfied. By (3.25) and (3.26), $v_i \in \{0, 1\}$. If $v_i = 0$, then by (3.28),

$$v_i = 0 \quad \Rightarrow \quad \theta_i = \theta_i(1 - v_i) \leq 0. \tag{3.29}$$

Since θ_i, the duration of the i^{th} subinterval, is non-negative, (3.29) implies that θ_i must be equal to zero. Thus, if $v_i = 0$, then $\theta_i = 0$ as required by (3.22b).

On the other hand, if $v_i = 1$, then by (3.27),

$$v_i = 1 \quad \Rightarrow \quad \epsilon_i - \theta_i = (\epsilon_i - \theta_i)v_i \leq 0 \quad \Rightarrow \quad \theta_i \geq \epsilon_i.$$

Hence, if $v_i = 1$, then $\theta_i \geq \epsilon_i$ as required by (3.22a). This completes the proof. □

Remark 3.1. Recall that constraints (3.12a) ensure that each active subsystem operates for at least the length of its minimum duration. We now show that, with the new constraints (3.27) and (3.28) in place, constraints (3.12a) actually become redundant. Suppose that (3.12b) and (3.25)-(3.28) are satisfied and $0 < \theta_i < \epsilon_i$. If $v_i = 0$, then (3.12b) and (3.28) imply $\theta_i = 0$, which is a contradiction. If $v_i = 1$, then (3.27) implies $\epsilon_i - \theta_i \leq 0$, which is also a contradiction. Thus, when (3.12b) and (3.25)-(3.28) hold, it is impossible for θ_i to lie in the open interval $(0, \epsilon_i)$. This implies that (3.12a) is redundant.

With this remark in mind, we now define a new problem as follows: Choose $(\boldsymbol{v}, \boldsymbol{\theta}, \boldsymbol{\zeta}) \in \mathbb{R}^N \times \mathbb{R}^N \times \mathbb{R}^r$ to minimize the cost function

$$\widetilde{G}_0(\boldsymbol{v}, \boldsymbol{\theta}, \boldsymbol{\zeta}) = \sum_{i=1}^{N} v_i \Phi_{0,i}(\widetilde{\boldsymbol{x}}(i|\boldsymbol{v}, \boldsymbol{\theta}, \boldsymbol{\zeta}), \boldsymbol{\zeta})$$

subject to the dynamics given by (3.9) and (3.17), the jump conditions given by (3.23), the constraints given by (3.12b) and (3.25)-(3.28), the canonical constraints given by (3.24) and the bounds given by (3.4). We refer to this problem as Problem 3C.

By transforming Problem 3B into Problem 3C, we obtain smooth jump conditions and smooth canonical functions. The remaining difficulty is that, as the constraints (3.25) and (3.26) define a disconnected region for $v_i, i = 1, \ldots, N$, standard numerical optimization algorithms will struggle to find an optimal solution. In the next section, we introduce a penalty method to overcome this difficulty.

3.3.3 An Exact Penalty Method

The exact penalty approach involves forming a new objective function by adding terms based on the constraints to the objective. With this approach, Problem 3C, a constrained optimization problem, is transformed into an approximate unconstrained problem that can be readily solved using the optimal control software MISER 3.3. MISER 3.3 automatically calculates the gradient of the objective function using a numerical procedure that involves integrating a costate system backwards in time. For more details, see [52, 60].

The *constraint violation* is defined by:

$$\begin{aligned}\Delta(\boldsymbol{v},\boldsymbol{\theta},\boldsymbol{\zeta}) = &\sum_{j=1}^{q_e}\left(\widetilde{G}_j(\boldsymbol{v},\boldsymbol{\theta},\boldsymbol{\zeta})\right)^2 + \sum_{j=q_e+1}^{q}\left(\min\{0,\widetilde{G}_j(\boldsymbol{v},\boldsymbol{\theta},\boldsymbol{\zeta})\}\right)^2\\ &+\sum_{i=1}^{N}\left(\max\{0,g_i(v_i)\}\right)^2+\sum_{i=1}^{N}\left(\max\{0,h_i(v_i,\theta_i)\}\right)^2\\ &+\sum_{i=1}^{N}\left(\max\{0,H_i(v_i,\theta_i)\}\right)^2+(t(N)-T)^2.\end{aligned}$$

Note that $\Delta(\boldsymbol{v},\boldsymbol{\theta},\boldsymbol{\zeta}) = 0$ if and only if constraints (3.9b), (3.24) and (3.26)-(3.28) are satisfied.

Using the strategy introduced in [36, 55, 129, 131], an exact penalty function $\widehat{J}_\sigma(\boldsymbol{v},\boldsymbol{\theta},\boldsymbol{\zeta},\lambda)$ is defined as follows:

$$\widehat{J}_\sigma(\boldsymbol{v},\boldsymbol{\theta},\boldsymbol{\zeta},\lambda) = \begin{cases}\widetilde{G}_0(\boldsymbol{v},\boldsymbol{\theta},\boldsymbol{\zeta}), & \text{if } \lambda = 0 \text{ and } \Delta(\boldsymbol{v},\boldsymbol{\theta},\boldsymbol{\zeta}) = 0,\\ \widetilde{G}_0(\boldsymbol{v},\boldsymbol{\theta},\boldsymbol{\zeta}) + \lambda^{-\alpha}\Delta(\boldsymbol{v},\boldsymbol{\theta},\boldsymbol{\zeta}) + \sigma\lambda^\beta, & \text{if } \lambda > 0,\\ +\infty, & \text{otherwise,}\end{cases}$$

where

- λ is a new decision variable;
- $\sigma > 0$ is the penalty parameter;
- α and β are positive constants satisfying $1 \le \beta \le \alpha$.

The new decision variable λ is subject to the following bounds:

$$0 \leq \lambda \leq \widetilde{\lambda}, \tag{3.30}$$

where $\widetilde{\lambda} > 0$ is a small positive number.

We now define the following unconstrained penalty problem: Choose $(\boldsymbol{v}, \boldsymbol{\theta}, \boldsymbol{\zeta}) \in \mathbb{R}^N \times \mathbb{R}^N \times \mathbb{R}^r$ and $\lambda \in \mathbb{R}$ to minimize $\widehat{J}_\sigma(\boldsymbol{v}, \boldsymbol{\theta}, \boldsymbol{\zeta}, \lambda)$ subject to the dynamics given by (3.9a) and (3.17), the jump conditions (3.23) and the bounds given by (3.4), (3.12b), (3.25) and (3.30). We refer to this problem as Problem 3D.

Note that when the penalty parameter σ is large, the third term $\sigma\lambda^\beta$ in $\widehat{J}_\sigma$ forces λ to be small, thus causing the second term $\lambda^{-\alpha}\Delta(\boldsymbol{v}, \boldsymbol{\theta}, \boldsymbol{\zeta})$ to severely penalize any constraint violations. When the penalty parameter σ is sufficiently large, any solution of the penalty problem (i.e. Problem 3D) is also an optimal solution of Problem 3C [55, 129–131].

Problem 3D is an optimal parameter selection problem involving a switched impulsive system with fixed jump times. Such problems can be solved using the software package MISER 3.3, which is based on gradient-descent optimization techniques.

In the next section, we demonstrate the efficiency of the proposed method outlined in this chapter with a numerical example.

3.4 Discussion based on Numerical Results

We consider the shrimp farming problem formulated by Yu and Leung in [133]. The dynamics in this problem are described by the following differential equations:

$$\dot{x}_1(t) = -0.03x_1(t), \qquad x_1(0) = 4.0 \times 10^4, \tag{3.31}$$

$$\dot{x}_2(t) = 3.5 - 10^{-5}x_1(t)x_2(t), \qquad x_2(0) = 1, \tag{3.32}$$

where

- t is the time in weeks;
- $x_1(t)$ is the number of remaining shrimp at time t;
- $x_2(t)$ is the average weight of an individual shrimp in grams at time t.

Let N denote the number of shrimp harvests and let T denote the time of the final harvest. In this example, we take $T = 13.2$ weeks.

Let $\tau_i \in [0, T]$ denote the time of the i^{th} harvest, with $\tau_N = T$ referring to the final harvest time. Note that these harvest times satisfy the following ordering constraints:

$$0 = \tau_0 \leq \tau_1 \leq \tau_2 \leq \cdots \leq \tau_N = T. \tag{3.33}$$

Furthermore, let ζ_i denote the fraction of shrimp stock harvested at time τ_i. Then

$$0 \leq \zeta_i \leq 1, \qquad i = 1, \dots, N. \tag{3.34}$$

The first state variable, which represents the number of shrimp, is subject to the following jump conditions:

$$x_1(\tau_i^+) - x_1(\tau_i^-) = -\zeta_i x_1(\tau_i^-), \qquad \text{if} \quad i \in \{1, \dots, N\} \text{ and } \tau_i - \tau_{i-1} \geq \epsilon_i, \tag{3.35}$$

where $\epsilon_i > 0$ is the given minimum duration between successive harvests. However, the second state variable, the average weight of shrimp, does not experience state jumps:

$$x_2(\tau_i^+) - x_2(\tau_i^-) = 0, \quad i = 1, \dots, N. \tag{3.36}$$

The following model is suggested by Yu and Leung [133] for the total revenue over the production cycle of 13.2 weeks:

$$R_1 = \sum_{i=1}^{N} \big((8 \times 10^{-3}) x_1(\tau_i^-) x_2(\tau_i) \zeta_i - 50\big). \tag{3.37}$$

The above expression for total revenue assumes a sale price of \$8 per kilogram for the shrimp and a fixed cost of \$50 per harvest. Note that expression (3.37) for the revenue will result in the fixed cost of \$50 dollars being imposed at every harvest time, even if multiple harvest times coincide. Thus, we consider the modified revenue function given by

$$R_2 = \sum_{i=1}^{N} \widehat{\Phi}_{0,i}(\tau_i - \tau_{i-1}, x_1(\tau_i^-), x_2(\tau_i), \zeta_i), \tag{3.38}$$

where $\widehat{\Phi}_{0,i}(\tau_i - \tau_{i-1}, x_1(\tau_i^-), x_2(\tau_i), \zeta_i)$ is defined as:

$$\widehat{\Phi}_{0,i}(\tau_i - \tau_{i-1}, x_1(\tau_i^-), x_2(\tau_i), \zeta_i) = \begin{cases} (8 \times 10^{-3}) x_1(\tau_i^-) x_2(\tau_i) \zeta_i - 50, & \text{if } \tau_i - \tau_{i-1} \geq \epsilon_i, \\ 0, & \text{if } \tau_i - \tau_{i-1} = 0 \end{cases} \tag{3.39}$$

We assume here that $N = 10$ (9 intermediate harvests and 1 final harvest).

The problem is to minimize $J = -R_2$, where R_2 is given by equation (3.38), subject to the dynamics (3.31)-(3.32), the jump conditions (3.35)-(3.36), the bounds (3.34) and the ordering constraints on τ_i given by (3.33). We first solve this problem using the method (i.e. the old method) in [53], which is implemented in MISER 3.3 and involves maximizing the revenue function R_1. Here, we impose a lower bound of 0.45 weeks on each of the durations between two consecutive harvests. Note that we are forced to do this due to the nature of the problem class in [53] where zero subsystem durations are not allowed.

The results obtained are tabulated in Table 3.1. The optimal value of the revenue function corresponding to the results depicted in Table 3.1 is 2.9037867×10^3. Note that the solution in Table 3.1 includes every potential harvest (i.e. no harvests have been deleted). This is

Harvest No.	x_1	x_2	Harvest Time	Harvesting Fraction
1	32173	6.86360	3.23733	0.113651
2	27769	7.70245	3.94938	0.118238
3	23778	8.67658	4.72935	0.123459
4	20174	9.81421	5.58682	0.129454
4	16933	11.15094	6.53325	0.136460
6	14033	12.73232	7.58251	0.144774
7	11452	14.61674	8.75118	0.154842
8	9168	16.88034	10.05957	0.167346
9	7163	19.62287	11.53240	0.183366
10	6814	22.97670	13.20000	1.000000

Table 3.1: Results from MISER3.3 using the old method

Harvest No.	x_1	x_2	Harvest Time	Harvesting Fraction
1	21535	7.77148	4.27134	0.388030
2	10580	12.64344	7.80877	0.453674
3	9000	22.23958	13.20000	1.000000

Table 3.2: Results from the new method described in this chapter

expected, as the algorithm by the old method has no capacity to eliminate non-optimal harvests. Thus, the solution in Table 3.1 may or may not be optimal.

Table 3.2 shows the results obtained using our new algorithm developed in this chapter. The notable feature here is that, by imposing a minimum duration of 0.45 weeks between two consecutive harvests, we obtained a maximum revenue of 3.1889334×10^3 with only 3 harvests taking place. In essence, in this example, 7 harvests have been removed (recall that we allowed up to $N = 10$ potential harvests).

The optimal harvest times corresponding to the solution in Table 3.2 are as follows:

$$\tau_1 = 4.27134, \quad \tau_2 = 4.27134, \quad \tau_3 = 4.27134, \quad \tau_4 = 4.27134, \quad \tau_5 = 7.80877,$$

$$\tau_6 = 7.80877, \quad \tau_7 = 7.80877, \quad \tau_8 = 7.80877, \quad \tau_9 = 7.80877, \quad \tau_{10} = 13.2.$$

Table 3.3 shows the progression of the penalty method corresponding to the solution in Table 3.2. Note that these results show a clear convergence of the objective function and the constraint violation as the penalty parameter σ is increased.

Figure 3.1 shows the number of shrimps at each point in the production cycle for the solutions in Tables 3.1 and 3.2. Note the difference in the solutions in Figure 3.1, one of which was obtained using the old method ([53]) and the other using our new method. As there are no jump conditions imposed on the average weight of the shrimp (see (3.36)), the two trajectories depicted in Figure 3.2 are almost the same.

σ	λ^*	Penalty Function Value	Constraint Violation
10^0	1.00000×10^{-1}	3.19832213×10^3	8.80240×10^{-2}
10^1	1.00000×10^{-1}	3.19806847×10^3	9.10594×10^{-2}
10^2	1.00000×10^{-1}	3.19553193×10^3	3.03930×10^{-3}
10^3	1.18314×10^{-3}	3.18890576×10^3	1.89485×10^{-9}
10^4	3.53827×10^{-6}	3.18893349×10^3	1.87380×10^{-16}
10^5	1.75704×10^{-6}	3.18893339×10^3	4.62264×10^{-17}
10^6	3.92612×10^{-7}	3.18893340×10^3	0.00000

Table 3.3: Numerical convergence using $\alpha = 2.00$ and $\beta = 1.55$

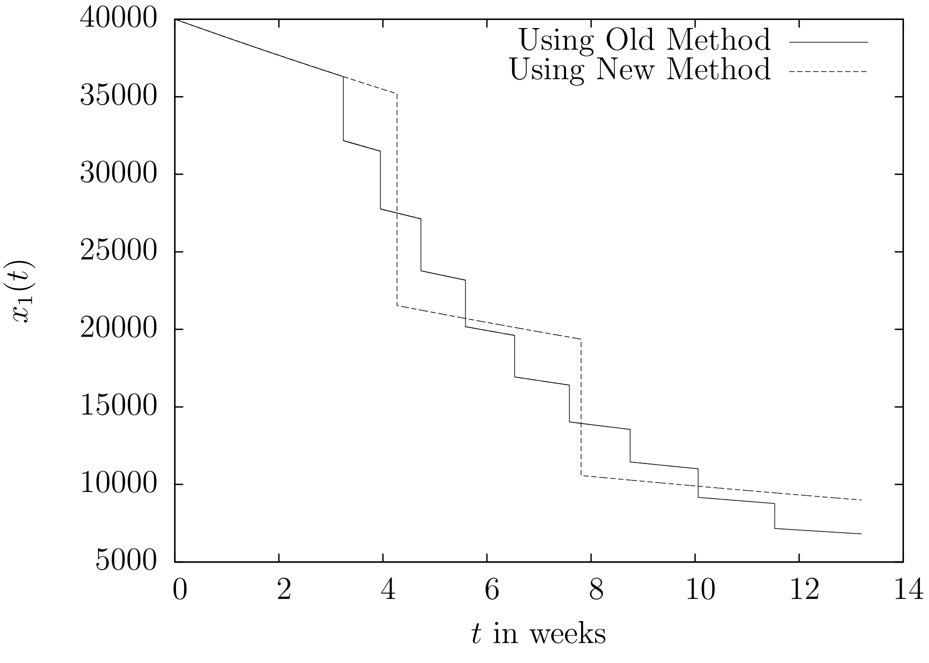

Figure 3.1: Number of shrimps

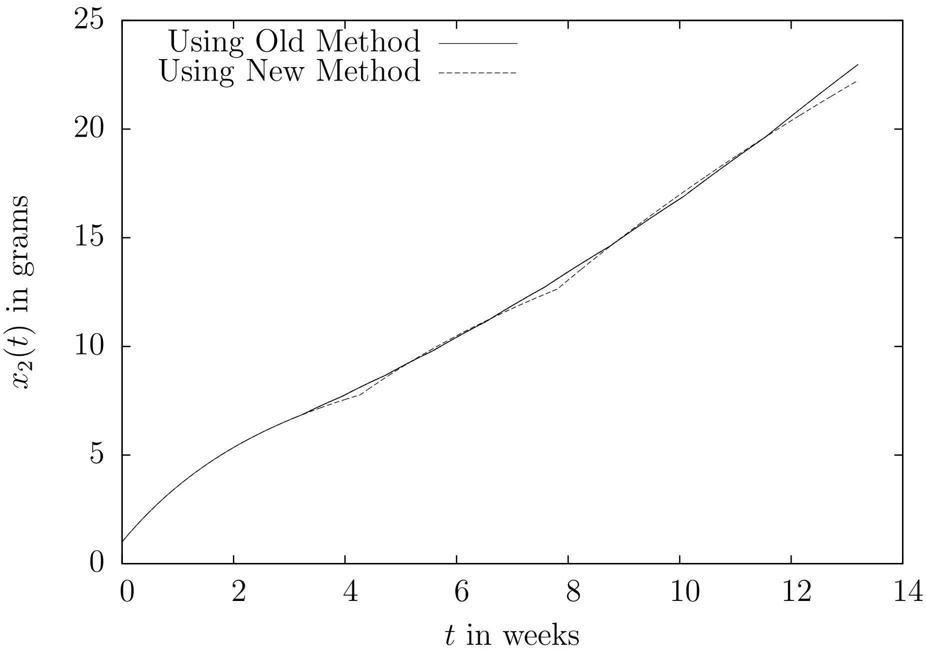

Figure 3.2: Average weight of shrimp

We have developed an efficient computational technique for solving a class of impulsive switched system optimal control problems, where the objective and constraints depend on the duration of each subsystem. This technique is based on a novel combination of the time-scaling transformation, binary relations, and exact penalty methods. The algorithm was successfully tested on a realistic shrimp farming problem.

CHAPTER 4

Dual–Mode Hybrid Systems with State–Dependent Switching Conditions

4.1 Introduction

The term *hybrid* is used to characterize systems that combine time–driven and event–driven dynamics, while the time–driven dynamics are typically defined by differential equations. The event–driven dynamics typically trigger switches in the time–driven dynamics. Generally, there are two types of switchings that can occur: either autonomous switchings or controlled switchings. Autonomous switching occurs automatically when the continuous state trajectory crosses from one region to another; controlled switchings are triggered externally (for example, by changing a gear). Switched hybrid systems are found in many practical applications such as medical science [71, 72, 128], biological systems [21], energy and power [93], chemical processes [30], manufacturing systems [13], laboratory plant [106] and robotics [8]. In this chapter, we solve a class of optimal control problems involving a dual–mode hybrid system. Here the autonomous switches are triggered by the state of the system moving between two different regions of the state space.

R. B. Martin [71, 72] formulated and solved an optimal control problem for the drug scheduling of cancer chemotherapy. This problem fits into the class of dual–mode hybrid system problems described above. The aim in this problem is to minimize the cancer tumour size by optimizing the administration of anti-cancer drugs. Constraints involving multiple characteristic times are imposed to ensure that the size of the tumour decreases at, or faster than, a specified rate. The two dynamic modes in the cancer chemotherapy problem are generated by the drug concentration being either below or above a given threshold value. Thus, this problem has dynamics which involve autonomous switchings from one region to the other. The problem formulation in [71] focuses on the central issue of decreasing the tumour size (subject to constraints imposed on the system). The control parameterization technique was used sequentially for an increasing number of partitions, and the problem was thus solved using nonlinear programming software developed by Schittkowski [98].

The authors of [65] solved the same cancer chemotherapy problem defined in [71,72] as an example to illustrate their solution method for a general class of optimal control problems, where the objective and certain constraints imposed on the system depend on two or more fixed time points called characteristic times. Control parameterization and a time–scaling transformation were used to obtain an approximate version of the original optimal control problem. Thereafter, standard gradient–based optimization techniques were used to solve the approximate problem. In particular, the subroutine NLPQLP [98] was used to solve the numerical example pertaining to cancer chemotherapy. The solution technique here is very similar to that of [71] with the additional feature of the well known time–scaling transformation being incorporated. The problem formulation in [65] ignores the issue of a hybrid dynamical system with the crucial aspect of the state–dependent switchings being omitted.

In [93], an optimal control problem involving a multi-mode hybrid power system was considered. Three distinct operations of mode arise due to the levels of supply from renewable and non-renewable sources, relative to the power demand level. Thus, the right-hand side of the dynamics are actually discontinuous with respect to the state which violates one of their key assumptions. The problem involves minimizing the operational cost of the hybrid power system while meeting the load requirements and satisfying other constraints imposed on the system. The time–scaling transformation is applied [45, 46] with a limited number of switchings allowed.

In [65, 71, 93], the authors effectively ignored the existence of autonomous switches in their dynamics. These problems were simply coded into the software in terms of conditional statements without any attempt to identify the exact switching times of the system.

More recently in [8], the objective functional gradients for a class of problems involving switched mode hybrid systems (with autonomous switches) were derived. A gradient-descent algorithm was then used to solve the problem. The approach in [8] is illustrated with a dual–mode hybrid system involving an autonomous mobile robot, which was required to reach a target (starting from a given initial position) while avoiding an obstacle along the way.

The approach in [30] which was used to solve problems involving hybrid systems, considers every solution of a Bellman type inequality which gives a lower bound on the optimal value function. A discretization of this *hybrid Bellman inequality* leads to a convex optimization problem. From the solution of the discretized problem, a value function that preserves the lower bound property is constructed. This then leads to a solution of an approximation of the optimal control problem. It is worth noting here that no constraints were imposed on the system in [30]. However, most practical applications of hybrid systems would involve some form of constraint.

As one reviews the literature on hybrid systems (see [8, 13, 30, 61, 79, 93, 106–108, 123, 128, 135]) over the past two decades, one comes across various approaches and solution methods with respect to different classes within the context of hybrid systems. The doctoral thesis authored by B. Passenburg [79] provides a good overview of various types of hybrid systems in existence. Moreover, it focuses on theoretical results and algorithms for indirect methods in optimal control of hybrid systems as opposed to direct methods and dynamic programming. Other solution methods discussed in the literature include mixed integer programming [106, 107], maximum principle [108] and branch and bound [79]. The solution technique proposed in [123] is another alternative method for hybrid systems. More specifically, it is based on a bi–level optimization method and also encompasses the concepts of constraint transcription and penalty functions to handle the continuous–time constraints imposed on the system.

Inspired by the work in [49, 129–132] on penalty methods, we propose a similar, but an alternative strategy for solving a class of *hybrid* optimal control problems involving a variety of complex constraints (a general form of canonical constraints involving multiple-characteristic times and continuous inequality constraints), in this chapter. To cast the problem into a form suitable for the exact penalty method, we need to apply a series of transcriptions and approximations as detailed below.

This chapter is organized as follows. In Section 4.2, we define the optimal control problem under consideration. In Subsection 4.3.1, we introduce a binary-valued function to govern the switching between the two dynamic modes. We then relax this binary-valued function to a continuous-valued function while imposing additional constraints to ensure the binary-valued requirement is met. In Subsection 4.3.2, we use the well-known control parameterization approach used in [14,57,58] to define an approximate finite dimensional version of the problem over a variable partition. In addition, we use the time–scaling technique adopted in [6, 50, 53] to transform the variable time points in this partition to fixed time points in a new time horizon. However, the constraints imposed on the resulting problem define a disconnected region, which poses a challenge for standard optimal control softwares such as MISER3.3 [34]. We adopt an exact penalty approach [129–131] in Section 4.4 to incorporate the constraints defining the disconnected region into the objective, so that the resulting problem can be readily solved by MISER3.3. Our numerical example in Section 4.5 also involves a special form of multiple-characteristic time constraints. We demonstrate that these constraints can be transformed using a transcription method into a standard form suitable for MISER3.3. This transcription technique entails the introduction of new auxiliary variables (or system parameters) as described in the latter part of that section. Our results show that the proposed method is effective and efficient.

4.2 Problem Formulation

Consider the following hybrid control system, defined on the time horizon $[0, T]$, in which the dynamics depend on the location of the state within the state space:

$$\dot{\boldsymbol{x}}(t) = \begin{cases} \boldsymbol{f}^1(\boldsymbol{x}(t), \boldsymbol{u}(t)), & \text{if } \boldsymbol{x}(t) \in \Omega_1, \\ \boldsymbol{f}^2(\boldsymbol{x}(t), \boldsymbol{u}(t)), & \text{if } \boldsymbol{x}(t) \in \Omega_2, \end{cases} \tag{4.1}$$

where

- $\boldsymbol{x}(t) \in \mathbb{R}^n$ is the state vector at time t;
- $\boldsymbol{u}(t) \in \mathbb{R}^r$ is the control vector at time t;
- Ω_1 and Ω_2 are given regions with disconnected interiors such that $\Omega_1 \cup \Omega_2 = \mathbb{R}^n$; and

- $\boldsymbol{f}^1, \boldsymbol{f}^2 : \mathbb{R}^n \times \mathbb{R}^r \to \mathbb{R}^n$ are given continuously differentiable functions.

Assumption 4.1. We assume that $\boldsymbol{f}^1(\boldsymbol{x}(t), \boldsymbol{u}(t)) = \boldsymbol{f}^2(\boldsymbol{x}(t), \boldsymbol{u}(t))$ when $\boldsymbol{x}(t) \in \Omega_1 \cap \Omega_2$.

According to equation (4.1), the system dynamics switch when the state moves from Ω_1 to Ω_2 or from Ω_2 to Ω_1. Thus, unlike the case for conventional switched systems, the switching times here are not independent decision variables, but are instead generated implicitly by the state trajectory.

We assume that Ω_1 and Ω_2 can be defined as follows:

$$\Omega_1 = \{\boldsymbol{x} \in \mathbb{R}^n : h(\boldsymbol{x}) \geq 0\}, \tag{4.2a}$$

$$\Omega_2 = \{\boldsymbol{x} \in \mathbb{R}^n : h(\boldsymbol{x}) \leq 0\}, \tag{4.2b}$$

where $h : \mathbb{R}^n \to \mathbb{R}$ is a given continuously differentiable function. Let the interiors of these regions be defined by:

$$\text{int}(\Omega_1) = \{\boldsymbol{x} \in \mathbb{R}^n : h(\boldsymbol{x}) > 0\},$$

$$\text{int}(\Omega_2) = \{\boldsymbol{x} \in \mathbb{R}^n : h(\boldsymbol{x}) < 0\}.$$

The initial conditions for system (4.1) are:

$$\boldsymbol{x}(0) = \boldsymbol{x}^0, \tag{4.3}$$

where $\boldsymbol{x}^0 \in \mathbb{R}^n$ is a given initial state.

Let

$$U = \left\{[u_1, \ldots, u_r]^\top \in \mathbb{R}^r : \alpha_j \leq u_j \leq \beta_j,\ j = 1, \ldots, r\right\}, \tag{4.4}$$

where α_j and β_j are given constants such that $\alpha_j < \beta_j$ for $j = 1, \ldots, r$. The set U is called the control restraint set. Any piecewise continuous function $\boldsymbol{u} : [0, T] \to \mathbb{R}^r$ such that $\boldsymbol{u}(t) \in U$ for all $t \in [0, T]$ is called an admissible control. Let $\mathcal{U}$ be the class of all such admissible controls.

For a given $\boldsymbol{u} \in \mathcal{U}$, let $\boldsymbol{x}(\cdot|\boldsymbol{u})$ be the corresponding state trajectory that satisfies the dynamic equations (4.1) and the initial condition (4.3).

We impose two types of constraints on system (4.1): canonical constraints and continuous inequality constraints. The canonical constraints (of both equality and inequality type) are expressed mathematically as:

$$G_v(\boldsymbol{u}) = \sum_{k=1}^{M} \Phi_{v,k}(\boldsymbol{x}(\tau_k|\boldsymbol{u})) + \int_0^T \mathcal{L}_v\left(\boldsymbol{x}(t|\boldsymbol{u}), \boldsymbol{u}(t)\right) dt \begin{cases} = 0, & v = 1, \ldots, p_e, \\ \geq 0, & v = p_e + 1, \ldots, p, \end{cases} \tag{4.5}$$

where $\Phi_{v,k} : \mathbb{R}^n \to \mathbb{R}$, $v = 1, \ldots, p$, $k = 1, \ldots, M$, and $\mathcal{L}_v : \mathbb{R}^n \times \mathbb{R}^r \to \mathbb{R}$, $v = 1, \ldots, p$, are continuously differentiable functions, and $\tau_1, \ldots, \tau_M$ are given characteristic times satisfying:

$$0 = \tau_0 < \tau_1 < \tau_2 < \cdots < \tau_M \leq T. \tag{4.6}$$

Note that the canonical form (4.5) can model many constraints of practical interest, such as interior-point constraints and terminal state constraints [52].

Continuous inequality constraints are expressed mathematically by:

$$g_v(\boldsymbol{x}(t|\boldsymbol{u})) \geq 0, \qquad t \in [0, T], \qquad v = 1, \ldots, q, \tag{4.7}$$

where $g_v : \mathbb{R}^n \to \mathbb{R}$, $v = 1, \ldots, q$, are given continuously differentiable functions. Such constraints impose restrictions on the system at every point in the time horizon.

We define a cost function in the same form as the canonical functions in (4.5):

$$G_0(\boldsymbol{u}) = \sum_{k=1}^{M} \Phi_{0,k}(\boldsymbol{x}(\tau_k|\boldsymbol{u})) + \int_0^T \mathcal{L}_0\left(\boldsymbol{x}(t|\boldsymbol{u}), \boldsymbol{u}(t)\right) dt, \tag{4.8}$$

where $\Phi_{0,k} : \mathbb{R}^n \to \mathbb{R}$, $k = 1, \ldots, M$, and $\mathcal{L}_0 : \mathbb{R}^n \times \mathbb{R}^r \to \mathbb{R}$ are given continuously differentiable functions.

Our problem is to choose an admissible control $\boldsymbol{u} \in \mathcal{U}$ to minimize $G_0(\boldsymbol{u})$ subject to the hybrid system dynamics (4.1), the initial condition (4.3), the canonical constraints (4.5) and the continuous inequality constraints (4.7). We refer to this problem as Problem 4A. We assume throughout this chapter that a solution exists for Problem 4A.

4.3 Problem Approximation

4.3.1 Reformulation of Dynamics

The system dynamics in (4.1) are non-smooth because of the state–dependent switching mechanism. Although methods are available for optimizing systems with non-smooth switching dynamics, they are generally only applicable when the switching mechanism is time-dependent and not state–dependent. Therefore, we introduce a new binary-valued decision function $z(t)$ defined as follows:

$$z(t) = \begin{cases} 1, & \text{if } \boldsymbol{x}(t) \in \text{int}(\Omega_1), \quad (4.9a) \\ 0, & \text{if } \boldsymbol{x}(t) \in \text{int}(\Omega_2), \quad (4.9b) \\ 0 \text{ or } 1, & \text{if } \boldsymbol{x}(t) \in \Omega_1 \cap \Omega_2. \quad (4.9c) \end{cases}$$

Using $z(t)$, we can write (4.1) as

$$\dot{\boldsymbol{x}}(t) = z(t)\boldsymbol{f}^1(\boldsymbol{x}(t), \boldsymbol{u}(t)) + (1 - z(t))\boldsymbol{f}^2(\boldsymbol{x}(t), \boldsymbol{u}(t)), \quad t \in [0, T]. \tag{4.10}$$

Note that equation (4.10) is a smooth reformulation of the state dynamics in (4.1). The authors of [73] present a method for solving an optimal control problem with an autonomous switched system. In Section 5.2 of their paper, they use a similar approach to convert state dynamics to a smooth form. Likewise, authors of [118] use "embedding functions" to transform the state dynamics to a smooth form. While this approach is very similar to the use of the binary–valued function above, the rest of their methodology is quite different to the approach taken in this chapter.

Standard optimization algorithms such as interior-point methods and sequential quadratic programming cannot handle binary constraints in the form of (4.9). Therefore, to proceed, we drop the binary requirements and consider each $z(t)$ as a continuous-valued function subject to the following constraints:

$$0 \leq z(t) \leq 1, \qquad t \in [0, T]. \tag{4.11}$$

Let $\mathcal{Z}$ denote the set of all piecewise continuous functions defined on $[0, T]$ and satisfying (4.11). Furthermore, let $\boldsymbol{x}(\cdot|\boldsymbol{u}, z)$ denote the solution of (4.10) and (4.3) corresponding to $(\boldsymbol{u}, z) \in \mathcal{U} \times \mathcal{Z}$. To ensure that $z(t)$ satisfies (4.9), we impose the following additional constraints:

$$H_1(z(t)) = z(t)(z(t) - 1) \geq 0, \qquad t \in [0, T], \tag{4.12}$$

$$H_2(\boldsymbol{u}, z(t)) = z(t)h(\boldsymbol{x}(t|\boldsymbol{u}, z)) - (1 - z(t))h(\boldsymbol{x}(t|\boldsymbol{u}, z)) \geq 0, \qquad t \in [0, T]. \tag{4.13}$$

The following theorem shows that constraint (4.11), together with (4.12) and (4.13), ensures that (4.9) is satisfied.

Theorem 4.1. *Let $z \in \mathcal{Z}$ be a given function. Then z satisfies* (4.9) *if and only if z satisfies* (4.11)-(4.13).

Proof. If z satisfies (4.9), then (4.11)-(4.13) are clearly satisfied. Thus, we focus on the reverse implication. Suppose that (4.11)-(4.13) hold true. Then it follows from (4.12) that either $z(t) \geq 1$ or $z(t) \leq 0$ for all $t \in [0, T]$. But we already know from (4.11) that $0 \leq z(t) \leq 1$ for each $t \in [0, T]$. Thus, we must have $z(t) \in \{0, 1\}$ for each $t \in [0, T]$.

If $z(t) = 1$, then it follows from (4.13) that $h(\boldsymbol{x}(t)) \geq 0$ and thus $\boldsymbol{x}(t) \in \Omega_1$ as required by (4.9). Similarly, if $z(t) = 0$, then (4.13) implies that $h(\boldsymbol{x}(t)) \leq 0$ and thus, $\boldsymbol{x}(t) \in \Omega_2$. This shows that (4.11)-(4.13) implies (4.9). □

We can now re-write the cost function (4.8) as:

$$G_0(\boldsymbol{u}, z) = \sum_{k=1}^{M} \Phi_{0,k}(\boldsymbol{x}(\tau_k|\boldsymbol{u}, z)) + \int_0^T \mathcal{L}_0\left(\boldsymbol{x}(t|\boldsymbol{u}, z), \boldsymbol{u}(t)\right) dt. \tag{4.14}$$

Similarly, the canonical constraints (4.5) and the continuous inequality constraints (4.7) can be written as:

$$\begin{aligned} G_v(\boldsymbol{u}, z) \quad &= \quad \sum_{k=1}^{M} \Phi_{v,k}(\boldsymbol{x}(\tau_k|\boldsymbol{u}, z)) \\ &\quad + \int_0^T \mathcal{L}_v\left(\boldsymbol{x}(t|\boldsymbol{u}, z), \boldsymbol{u}(t)\right) dt \begin{cases} = 0, & v = 1, \ldots, p_e, \\ \geq 0, & v = p_e + 1, \ldots, p. \end{cases} \end{aligned} \tag{4.15}$$

and

$$g_v(\boldsymbol{x}(t|\boldsymbol{u}, z)) \geq 0, \qquad t \in [0, T], \qquad v = 1, \ldots, q, \tag{4.16}$$

respectively.

On the basis of Theorem 1, it is clear that Problem 4A is equivalent to the following problem: Choose a pair $(\boldsymbol{u}, z) \in \mathcal{U} \times \mathcal{Z}$ to minimize the cost function (4.14), subject to the system (4.10), the initial condition (4.3) and the constraints (4.15), (4.16), (4.11), (4.12) and (4.13). We refer to this problem as Problem 4B.

4.3.2 Control Parameterization

Define a partition on the time interval $[0, T]$ as follows:

$$P = \{t_0, t_1, \ldots, t_N\}, \tag{4.17}$$

where $t_0 = 0$, $t_N = T$ and $t_{j-1} \leq t_j$ for each $j = 1, \ldots, N$. We assume that P is chosen so that for each $i \in \{1, \ldots, M\}$, there exists a corresponding $\nu_i \in \{1, \ldots, N\}$ such that $t_{\nu_i} = \tau_i$, i.e., the partition includes the characteristic times in the constraints (4.15). An example of how the $\nu_i, i = 1, \ldots, M$, are chosen is given in the numerical example in Section 4.5.

We now approximate the control $\boldsymbol{u} \in \mathcal{U}$ by a piecewise constant function consistent with the partition defined in (4.17). In other words, $t_j, j = 1, \ldots, N-1$, are the switching points of the piecewise constant control function. We allow these switching points to be optimization variables. Although this increases the accuracy of the approximation, it makes the problem more difficult to solve, as variable switching times cause numerical difficulties as documented in [63]. Hence, we will apply the well-known time–scaling transformation described in [5,50,53] to transform the variable switching times into fixed time points in a new time horizon. Define

$$\theta_j = t_j - t_{j-1}, \quad j = 1, \ldots, N. \tag{4.18}$$

The time–scaling transformation is achieved by introducing a new time variable $s \in [0, N]$, and relating s to t through the following initial value problem,

$$\dot{t}(s) = \theta_j, \qquad t(0) = 0, \tag{4.19a}$$

subject to the constraint

$$t(N) = T. \tag{4.19b}$$

To ensure that the characteristic times are transformed appropriately, we also require the following constraints in addition to (4.19b):

$$t(\nu_i) = \tau_i, \quad i = 1, \ldots, M. \tag{4.20}$$

For $s \in [k-1,\ k]$, integration of (4.19a), along with (4.18), yields

$$t(s) = \sum_{j=1}^{k-1} \theta_j + \theta_k (s - k + 1). \tag{4.21}$$

Let Θ be the set of all $\boldsymbol{\theta} = [\theta_1, \theta_2, \ldots, \theta_N]^\top \in \mathbb{R}^N$ satisfying $\theta_j \geq 0$ for all $j = 1, \ldots, N$.

Further, let $\chi_{[j-1,j)}(s)$ be the indicator function defined by

$$\chi_{[j-1,\ j)}(s) = \begin{cases} 1, & \text{if } s \in [j-1, j), \\ 0, & \text{otherwise.} \end{cases}$$

The approximate control can now be written as:

$$\widetilde{\boldsymbol{u}}(s) = \sum_{j=1}^{N} \boldsymbol{\sigma}_j \chi_{[j-1,j)}(s), \tag{4.22}$$

where

$$\boldsymbol{\sigma}_j \in U, \quad j = 1, \ldots, N. \tag{4.23}$$

We also assume that $\widetilde{\boldsymbol{z}}$ is binary–valued and is piecewise constant with respect to the partition defined in (4.17). Hence, we may approximate it by

$$\widetilde{z}(s) = \sum_{j=1}^{N} \xi_j \chi_{[j-1,j)}(s) \tag{4.24}$$

where

$$\xi_j \in [0, 1], \quad j = 1, \ldots, N. \tag{4.25}$$

We define $\boldsymbol{\sigma} = [(\boldsymbol{\sigma}_1),^\top \ldots, (\boldsymbol{\sigma}_N)^\top]^\top$ and $\boldsymbol{\xi} = [\xi_1, \xi_2, \ldots, \xi_N]^\top$.

Let Σ be the set of all vectors $\boldsymbol{\sigma}$ and Υ be the set of all vectors $\boldsymbol{\xi}$ such that their components satisfy (4.23) and (4.25), respectively. In the new time horizon, (4.10) is now transformed to:

$$\dot{\widetilde{\boldsymbol{x}}}(s) = \theta_j \big(\xi_j \boldsymbol{f}^1(\widetilde{\boldsymbol{x}}(s), \boldsymbol{\sigma}_j) \ + \ (1 - z_j) \boldsymbol{f}^2(\widetilde{\boldsymbol{x}}(s), \boldsymbol{\sigma}_j)\big), \quad s \in [j-1, j), \ j = 1, \ldots, N. \tag{4.26}$$

Let $\widetilde{\boldsymbol{x}}(\cdot|\boldsymbol{\sigma}, \boldsymbol{\xi}, \boldsymbol{\theta})$ be the solution of (4.26) and (4.3) corresponding to $(\boldsymbol{\sigma}, \boldsymbol{\xi}, \boldsymbol{\theta}) \in \Sigma \times \Upsilon \times \Theta$.

Therefore, our transformed cost function now becomes:

$$\widetilde{G}_0(\boldsymbol{\sigma}, \boldsymbol{\xi}, \boldsymbol{\theta}) = \sum_{k=1}^{M} \Phi_{0,k}(\widetilde{\boldsymbol{x}}(\nu_k|\boldsymbol{\sigma}, \boldsymbol{\xi}, \boldsymbol{\theta})) + \sum_{j=1}^{N} \int_{j-1}^{j} \theta_j \mathcal{L}_0 \left(\widetilde{\boldsymbol{x}}(s|\boldsymbol{\sigma}, \boldsymbol{\xi}, \boldsymbol{\theta}), \boldsymbol{\sigma}_j\right) ds. \tag{4.27}$$

Furthermore, the canonical constraints in (4.5) are transformed to:

$$\begin{aligned} \widetilde{G}_v(\boldsymbol{\sigma}, \boldsymbol{\xi}, \boldsymbol{\theta}) \ &= \ \sum_{k=1}^{M} \Phi_{v,k}(\widetilde{\boldsymbol{x}}(\nu_k|\boldsymbol{\sigma}, \boldsymbol{\xi}, \boldsymbol{\theta})) \\ &\quad + \sum_{j=1}^{N} \int_{j-1}^{j} \theta_j \mathcal{L}_v \left(\widetilde{\boldsymbol{x}}(s|\boldsymbol{\sigma}, \boldsymbol{\xi}, \boldsymbol{\theta}), \boldsymbol{\sigma}_j\right) ds \begin{cases} = 0, & v = 1, \ldots, p_e, \\ \geq 0, & v = p_e + 1, \ldots, p. \end{cases} \end{aligned} \tag{4.28}$$

The continuous inequality constraints in (4.7) become

$$g_v(\widetilde{\boldsymbol{x}}(s|\boldsymbol{\sigma}, \boldsymbol{\xi}, \boldsymbol{\theta})) \geq 0, \qquad s \in [j-1, j), \qquad v = 1, \ldots, q. \tag{4.29}$$

By the same analogy, (4.12) and (4.13) will be transformed into:

$$H_1(\xi_j) = \xi_j(\xi_j - 1) \geq 0, \qquad s \in [j-1, j), \quad j = 1, \ldots, N, \tag{4.30}$$

and

$$\begin{aligned} H_2(s) = \xi_j h(\widetilde{\boldsymbol{x}}(s|\boldsymbol{\sigma}, \boldsymbol{\xi}, \boldsymbol{\theta})) - (1 - \xi_j) h(\widetilde{\boldsymbol{x}}(s|\boldsymbol{\sigma}, \boldsymbol{\xi}, \boldsymbol{\theta})) \geq 0, \\ s \in [j-1, j), \quad j = 1, \ldots, N. \end{aligned} \tag{4.31}$$

We now define an approximate problem as follows: Choose $(\boldsymbol{\sigma}, \boldsymbol{\xi}, \boldsymbol{\theta}) \in \Sigma \times \Upsilon \times \Theta$ to minimize the cost function given by (4.27) subject to the system represented by (4.26), the initial condition (4.3), the canonical constraints (4.28), the continuous inequality constraints (4.29), (4.30), (4.31) as well as constraints (4.19b) and (4.20). We refer to this problem as Problem 4C. Note that this is an approximation of Problem 4B.

Standard numerical optimization algorithms will struggle to find an optimal solution for Problem 4C as the constraint (4.30) defines a disconnected region. In the next section, we introduce a penalty method to overcome this difficulty.

4.4 Exact Penalty Method

We adopt the exact penalty approach in [129–131], to transform Problem 4C into an unconstrained optimization problem that can be easily solved using the optimal control software MISER 3.3.

The *constraint violation* is defined by:

$$\begin{aligned} \Delta(\boldsymbol{\sigma}, \boldsymbol{z}, \boldsymbol{\theta}) = \sum_{v=1}^{p_e} \left(\widetilde{G}_v(\boldsymbol{\sigma}, \boldsymbol{\xi}, \boldsymbol{\theta})\right)^2 + \sum_{v=p_e+1}^{p} \left(\min\{0, \widetilde{G}_v(\boldsymbol{\sigma}, \boldsymbol{\xi}, \boldsymbol{\theta})\}\right)^2 \\ + \sum_{v=1}^{q} \int_0^N \left(\min\{0, g_v(\widetilde{\boldsymbol{x}}(s|\boldsymbol{\sigma}, \boldsymbol{\xi}, \boldsymbol{\theta}))\}\right)^2 ds + \sum_{j=1}^{N} \left(\min\{0, H_1(\xi_j)\}\right)^2 + \\ \sum_{j=1}^{N} \left(\min\{0, H_2(\xi_j)\}\right)^2 + \sum_{j=1}^{M} \left(t(\nu_j) - \tau_j\right)^2 + \left(t(N) - T\right)^2. \end{aligned}$$

Note that $\Delta(\boldsymbol{\sigma}, \boldsymbol{\xi}, \boldsymbol{\theta}) = 0$ if and only if the constraints (4.19b), (4.20) and (4.28)-(4.31) are satisfied.

Using the strategy introduced in [129–131], an exact penalty function $\widehat{J}_\delta(\boldsymbol{\sigma}, \boldsymbol{\xi}, \boldsymbol{\theta}, \epsilon)$ is defined as follows:

$$\widehat{J}_\delta(\boldsymbol{\sigma},\boldsymbol{\xi},\boldsymbol{\theta},\epsilon) = \begin{cases} \widetilde{G}_0(\boldsymbol{\sigma},\boldsymbol{\xi},\boldsymbol{\theta}), & \text{if } \epsilon = 0 \text{ and } \Delta(\boldsymbol{\sigma},\boldsymbol{\xi},\boldsymbol{\theta}) = 0, \\ \widetilde{G}_0(\boldsymbol{\sigma},\boldsymbol{\xi},\boldsymbol{\theta}) + \epsilon^{-\alpha}\Delta(\boldsymbol{\sigma},\boldsymbol{\xi},\boldsymbol{\theta},\epsilon) + \delta\epsilon^{\gamma}, & \text{if } \epsilon > 0 \\ +\infty, & \text{otherwise,} \end{cases} \tag{4.32}$$

where ϵ is a new decision variable, $\delta > 0$ is the penalty parameter and α and γ are positive constants satisfying $1 \leq \gamma \leq \alpha$.

The new decision variable ϵ is subject to the following bounds:

$$0 \leq \epsilon \leq \widetilde{\epsilon}, \tag{4.33}$$

where $\widetilde{\epsilon} > 0$ is a small positive number.

We now define the following unconstrained problem: Choose $(\boldsymbol{\sigma},\boldsymbol{\xi},\boldsymbol{\theta}) \in \Sigma \times \Upsilon \times \Theta$ and $\epsilon \in [0,\widetilde{\epsilon}]$ to minimize $\widehat{J}_\delta(\boldsymbol{\sigma},\boldsymbol{\xi},\boldsymbol{\theta},\epsilon)$ subject to the system (4.26) and the initial condition (4.3). We refer to this problem as Problem 4D.

Note that when the penalty parameter δ is large, the third term $\delta\epsilon^{\gamma}$ in $\widehat{J}_\delta$ forces ϵ to be small, thus causing the second term $\epsilon^{-\alpha}\Delta(\boldsymbol{\sigma},\boldsymbol{\xi},\boldsymbol{\theta})$ to severely penalize any constraint violations. When the penalty parameter δ is sufficiently large, any solution of the penalty problem (i.e. Problem 4D) is also an optimal solution of Problem 4C [129–131].

In the next section, we demonstrate the efficiency of the proposed method with a numerical example.

4.5 Numerical Results and Analysis

We consider the cancer chemotherapy problem addressed in [65, 71, 72].

Let $N(t)$ denote the number of cancer cells in the tumour at time t. In this example, we seek to minimize $-x_1(T)$, where $x_1(t)$ is given by $x_1(t) = \ln(\mu/N(t))$. The cancer treatment commences at time $t = 0$ and ends at time $t = T$.

The dynamics in this problem are described by the following differential equations:

$$\dot{x}_1(t) = \begin{cases} -\lambda x_1(t), & \text{if } x_2(t) - 10 \leq 0, \\ -\lambda x_1(t) \ + \kappa[x_2(t) - 10], & \text{if } x_2(t) - 10 \geq 0. \end{cases} \tag{4.34}$$

$$\dot{x}_2(t) = u(t) - \beta x_2(t). \tag{4.35}$$

Note that:

- t is the time in days;
- $x_1(t)$ is a function of $N(t)$, the number of cancer cells at time t;

- $x_2(t)$ is the concentration of the anti-cancer drug at the cancer site at time t;
- $u(t)$ is the rate of delivery of the drug at time t.

The initial conditions are $x_1(0) = \ln(100)$ and $x_2(0) = 0$.

The following restrictions are imposed on the anti-cancer drug concentration to ensure that toxins absorbed by body of the patient do not exceed the medical limits:

$$0 \leq x_2(t) \leq v_{max}, \qquad t \in [0, T], \tag{4.36}$$

and

$$\int_0^T x_2(s)ds \leq v_{cum}, \tag{4.37}$$

Following the lead of R. B. Martin [71], we impose the following restrictions on the tumour size to make it decrease at, or faster than a given rate:

$$N(\tau_i) \leq \rho N(\tau_{i-1}), \qquad i = 1, \ldots, M, \tag{4.38}$$

where

$$\tau_i = iT/4$$

and

$$0 < \rho < 1 \tag{4.39}$$

The restrictions (4.38) are mathematically equivalent to:

$$x_1(\tau_i) - x_1(\tau_{i-1}) + \ln \rho \geq 0, \qquad \text{for} \qquad i = 1, \ldots, M. \tag{4.40}$$

The model parameters used in this problem are: $T = 84.0$ days, $\lambda = 9.9 \times 10^{-4}$, $\kappa = 8.4 \times 10^{-3}$, $\beta = 0.27$, $v_{max} = 50.0$, $v_{cum} = 2100$, $\rho = 0.5$, $\mu = 10^{12}$ and $M = 3$.

Note that constraints (4.40) involve two characteristic times and are thus not in the canonical form (4.5) where the $\Phi_{v,k}$ function depends only on τ_k. We use the following transcription method to overcome this problem.

Define auxiliary variables w_1 and w_2 such that $w_1 = \tau_1$, $w_2 = \tau_2$. Then (4.40) may be written as:

$$x_1(\tau_1) - x_1(0) + \ln \rho \geq 0, \tag{4.41}$$

$$x_1(\tau_2) - x_1(w_1) + \ln \rho \geq 0, \tag{4.42}$$

$$x_1(\tau_3) - x_1(w_2) + \ln \rho \geq 0, \tag{4.43}$$

with the additional equality constraints

$$\tau_1 - w_1 = 0, \tag{4.44}$$

and

$$\tau_2 - w_2 = 0. \tag{4.45}$$

These auxiliary variables are effectively system parameters as discussed in Section 1.3.2 of this thesis. (4.41)–(4.45) are in the canonical form (4.5). Note that we also impose the

σ	ϵ^*	Objective Function	Number of Cancer Cells
10^1	1.71735×10^{-2}	-1.689154×10^1	4.614210×10^4
10^2	1.71735×10^{-2}	-1.689154×10^1	4.614210×10^4
10^3	1.71733×10^{-2}	-1.689154×10^1	4.614210×10^4
10^4	2.16421×10^{-3}	-1.688815×10^1	4.629879×10^4
10^5	2.16420×10^{-3}	-1.688814×10^1	4.629925×10^4
10^6	2.16420×10^{-3}	-1.688814×10^1	4.629925×10^4
10^7	2.16420×10^{-3}	-1.688814×10^1	4.629925×10^4
10^8	2.16420×10^{-3}	-1.688814×10^1	4.629925×10^4
10^9	2.16420×10^{-3}	-1.688814×10^1	4.629925×10^4

Table 4.1: Numerical convergence using $\alpha = 3.00, \gamma = 2.00$ and 32 control parameters

N	No of Cancer Cells
16	6.396×10^4
32	5.611×10^4
64	4.976×10^4
128	4.878×10^4
256	4.878×10^4

Table 4.2: Results from R. B. Martin's paper in *Automatica*, 1992

following bound constraints on w_1 and w_2:

$$0 \leq w_i \leq T, \quad i = 1, 2. \tag{4.46}$$

The problem is to minimize $J = -x_1(t)$, subject to the dynamics given by (4.34) and (4.35), the given initial conditions and constraints given by (4.36), (4.37), (4.40), (4.41)–(4.45) and (4.46). After transforming and approximating the problem as described in Sections 4.3 and 4.4, we solve the resulting unconstrained problem (we choose $N = 32$ for the partition) using MISER3.3. Results in Table 4.1 shows numerical convergence as the penalty parameter δ increases. The optimal value of ϵ is denoted by ϵ^* for each run.

While Martin [71] also used the technique of control parameterization to solve this problem, he used fixed partitions of time horizon as opposed to variable partitions. Results from Martin [71] for various cases of N are show in Table 4.2.

Note that the final number of cells $N(t)$ obtained from the algorithm in this chapter with $N = 32$ is 4.629925×10^4 which is smaller than the best result in [71] regardless of the partition size. The graph in Figure 4.3 indicates that the number of cancer cells are reduced significantly by the end of the treatment period. The corresponding drug concentration and drug delivery rate are shown in Figures 4.1 and 4.2 respectively.

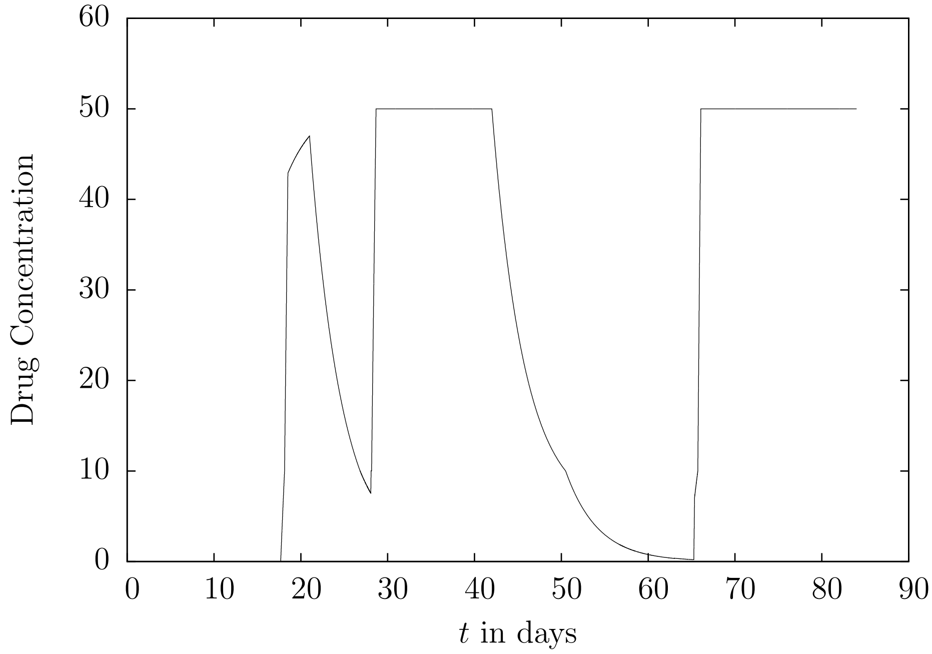

Figure 4.1: Drug concentration

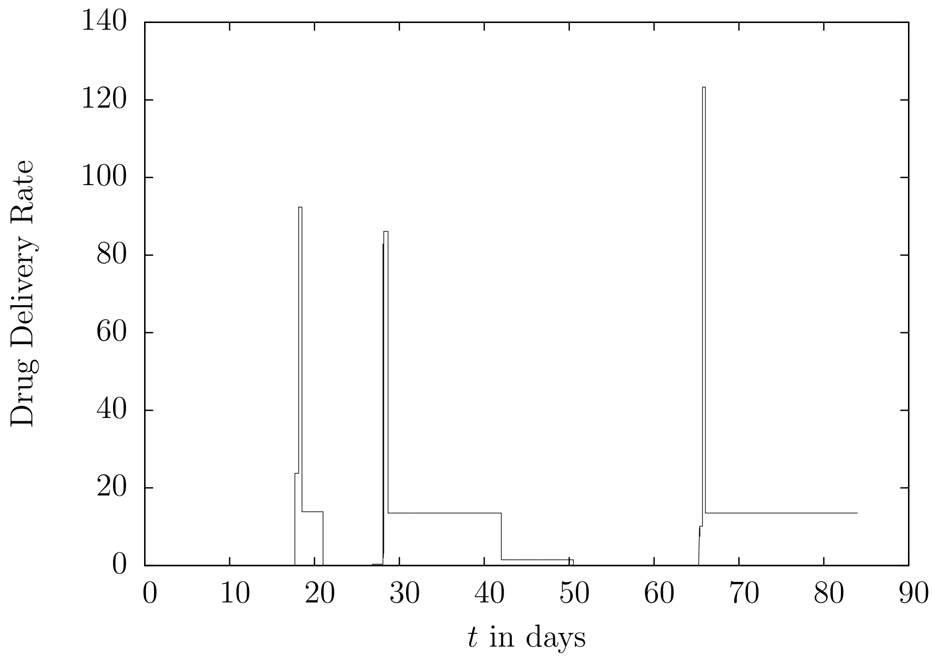

Figure 4.2: Drug delivery rate

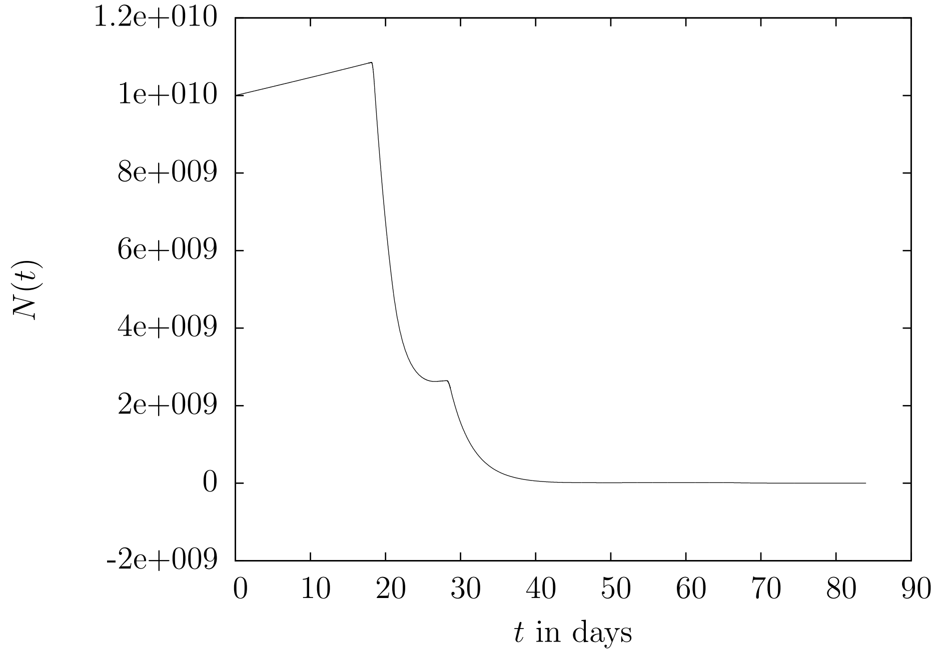

Figure 4.3: Number of tumor cells

4.6 Concluding Remarks

We have presented an efficient computational method for solving a class of optimal control problems involving a dual–mode hybrid system with state–dependent switching conditions subject to all time–constraints as well as multiple characteristic–time constraints. The method involves the introduction of binary variables to transcribe the dynamics, approximation by control parameterization, a time–scaling transformation and the use of an exact penalty method. Results show that the method is efficient and reliable.

CHAPTER 5

Conclusions

5.1 Main Contribution of this Thesis

Section 1.1 of this thesis is devoted to a discussion of the origination, development and advancement of the concepts of *calculus of variations*, *optimization* and *optimal control* during the last 350 years. Significant contributions by renowned mathematicians (for example, [3,82]), which brought about pivotal changes in the solution methods for optimization and optimal control problems, have been highlighted, while pursuing a chronological order of events throughout the section.

In the following sections, three numerical techniques, i.e. control parameterization, time-scaling transformation and the exact penalty approach, are described with respect to the mathematical formulation of a general optimal control problem. As the optimal control software MISER3.3 [34] was used to solve all of the numerical problems given in this thesis, it was only fitting that a description of the workings of this software be included in this section. This description has been based on a general formulation of an optimal control problem, which includes control variables, constraints involving individual characteristic times, system parameters and state jumps as detailed in Subsection 1.3.2. Furthermore, the gradient formulae and the computational strategies that MISER3.3 uses to solve such optimal control problems are stated explicitly to enhance the understanding of the reader, with the aim of explaining the intricate workings of MISER3.3 beyond what an ordinary user would see on the computer screen. The input data required by MISER3.3, which is specific to the problem being solved, must be supplied to MISER prior to running the program. Besides, practical issues which need to be understood by the user when using MISER3.3 to solve a problem are highlighted in Subsection 1.3.2.

Chapter 2 deals with a computational algorithm for a class of optimal control problems with discontinuous objective functions. This class of problems is motivated by a practical situation where shrimp need to be harvested at certain times. The price of shrimp is modelled as a piecewise–constant function depending on the shrimp weight. The problem of choosing the harvesting times to maximize total revenue turns out to be a discontinuous optimal control problem governed by an impulsive system. The proposed computational method comprises of the well–known time–scaling transformation and a smoothing transformation through the introduction of binary variables. These steps are followed by the exact penalty method, which transforms the problem into an approximate unconstrained problem which can be readily solved by MISER3.3. The problem solved here is an extension of the previous partial harvesting model of shrimp culture proposed by R. Yu and P.

Leung in [133] in which the price per kilogram of shrimp was assumed to be fixed. Note that the discontinuous objective function in this thesis came about as a result of the more realistic price–size relationship of shrimp. This is a significant practical improvement on the original formulation. The price–size relationship of shrimp modelled in this problem is more viable in a commercial environment.

Reference [134] also considers the price–size relationship of shrimp in a partial harvesting situation. The authors of this paper used MS Excel spreadsheets to compare the net revenue from the shrimp sales, under various conditions, to work out the optimal harvesting policy. However, the computational approach used in Chapter 2 of this thesis is based on an impulsive switched system subject to dynamics comprising of the weight of the shrimp and the number of shrimp, as well as instantaneous jumps involving the number of shrimp. Through the development of an innovative algorithm that incorporates the exact penalty approach, the original problem is transformed into an unconstrained problem that was readily solvable by MISER3.3. As all of the factors affecting the net revenue from the shrimp sales are taken into consideration through the dynamics and the revenue model in a simultaneous fashion, this solution technique is more advanced than the approach in [134]. Since there is a lack of computational techniques for solving optimal control problems with discontinuous objective functions, the algorithm developed here presents a novel computational algorithm which combines the time–scaling transformation and an exact penalty method to solve the optimal control problem numerically.

Chapter 3 deals with another class of problems involving impulsive switched systems, where the state variables exhibit instantaneous state jumps. Such systems consist of multiple subsystems operating in succession, with possible instantaneous state jumps occurring when the system switches from one subsystem to another. The control variables are the subsystem durations and a set of system parameters influencing the state jumps. In contrast to most literature on the control of impulsive switched systems, we assume that, every potential subsystem need not be active during the time horizon, i.e. certain consecutive switching times may coincide. Thus, when such switching times do coincide, there will be only one switch instead of two or more switches. However, users of the traditional time–scaling transformation (for example, see [5, 14, 50]) have over–looked this important issue in previous papers. It may be optimal to delete certain subsystem durations in many problems and hence, we impose a restriction that any active subsystem must be active for a minimum non-negligible duration of time. This restriction leads to a disconnected region for the subsystem durations. The problem of minimizing a given cost function subject to canonical constraints, state jumps and restrictions of this nature is a non-standard optimal control problem that cannot be solved using conventional techniques. A computational algorithm is developed by combining the well–known time-scaling transformation with an exact penalty method. The exact penalty method is used to handle the binary constraints imposed on the system to ensure that there is exactly one state jump if multiple switching times coincide at a single point. The effectiveness of this algorithm is illustrated through a numerical example on the optimization of shrimp harvesting operations.

In Chapter 4, we develop a computational approach for optimizing a class of hybrid systems whose state dynamics switch between two distinct modes. For this class of systems, the times at which the mode transitions occur cannot be specified directly, but are instead governed by a state-dependent switching condition. The objective function is optimized subject to a set of canonical (inequality and equality) constraints as well as continuous inequality constraints. Furthermore, the objective and canonical constraint functions depend on individual characteristic times. By introducing an auxiliary binary–valued control function to represent the current mode, the hybrid system under consideration is transformed into a standard dynamic system subject to path constraints. This transformation procedure involving the transcription method, as well as the introduction of the binary–valued control function, overcomes the challenges encountered when solving such problems via standard optimal control software such as MISER3.3. However, the resulting variable switching times make this a non–standard optimal control problem. Hence, we combine the control parameterization technique together with the well–known time-scaling transformation and use an exact penalty method thereafter to transform the constrained problem into an approximate unconstrained problem. The resulting unconstrained problem is readily solved by MISER3.3. A numerical example on cancer chemotherapy is included to demonstrate the effectiveness of our proposed algorithm.

In summary, this thesis contains solution techniques to three different classes of non-standard optimal control problems. The time-scaling transformation and an exact penalty method were employed in each of the solution strategies, resulting in problems that can be readily solved by standard optimal control software such as MISER3.3.

5.2 Future Research Directions

The development of numerical algorithms for the three non-standard optimal control problems covered in this thesis has paved the way for unexplored avenues of research in the future. Possible extensions to the novel results and insights gained in this thesis are discussed below.

The numerical example used in Chapters 2 and 3 is based on the shrimp harvesting model considered by Yu and Leung [133] for a single batch production cycle. In Chapter 2, we use a piecewise constant price function which depends on the average shrimp weight and a fixed cost per harvest in the revenue function. In addition, we use the same equation as in [133] to factor in the shrimp mortality rate. Future studies on the shrimp harvesting problem should consider incorporating additional factors into the model, such as variable initial conditions, varying harvesting costs that are dependent on numerous factors (such as the wages of labourers, the number of labour hours, the weight of shrimp harvested, etc.) and the optimization of the feeding rate. References [33, 62, 86, 115] discuss the aspects of shrimp growth, mortality and migration and introduce various mathematical equations to model these factors. Examining shrimp harvesting models for multiple continuous–production cycles of shrimp together with suitable cost functions for varying harvesting costs (discussed in this paragraph) and considering other functions mentioned in [33,62,86,115] involving growth, mortality and migration of shrimp, provide potential avenues of future research.

In Chapter 2 of this thesis, a solution technique is developed for a class of problems with discontinuous objective functions. In this class of problems, apart from the ordering constraint on the jump times, there are no other constraints imposed on the system. Hence, it would be of great value to develop numerical algorithms to solve optimal control problems with a discontinuous objective function subject to a complex set of constraints such as canonical constraints, continuous inequality constraints and constraints involving multiple characteristic times. As science and technology keep advancing at a fast rate, the solution technique to such classes of problems are likely to have numerous practical applications.

Chapter 4 deals with a dual–mode hybrid system in which autonomous switchings occur. Note that the problem formulation in Section 4.2, where the state dynamics are defined differently in two regions Ω_1 and Ω_2 is such that $\boldsymbol{f}^1(\boldsymbol{x}(t), \boldsymbol{u}(t)) = \boldsymbol{f}^2(\boldsymbol{x}(t), \boldsymbol{u}(t))$ when $\boldsymbol{x}(t) \in \Omega_1 \cap \Omega_2$, where $\boldsymbol{f}^1$ and $\boldsymbol{f}^2$ represent the subsystem dynamics. It would be of interest to develop a computational method to solve an optimal control problem with a dual–mode hybrid system where $\boldsymbol{f}^1(\boldsymbol{x}(t), \boldsymbol{u}(t)) \neq \boldsymbol{f}^2(\boldsymbol{x}(t), \boldsymbol{u}(t))$ when $\boldsymbol{x}(t) \in \Omega_1 \cap \Omega_2$. As the class of problems considered in Chapter 4 involves a hybrid system with autonomous switchings, it would also be interesting to compare the different strategies which need to be undertaken to solve a problem involving a hybrid system with externally triggered switchings. In addition, further research would be valuable to develop solution techniques to event–driven hybrid problems with state jumps, as this feature would pose a challenge. Future work should also be aimed at hybrid systems with more than two modes (see [93]), as we often encounter these types of problems in real life.

Further, research should be undertaken which would allow for different penalty weights for individual constraints in the exact penalty approach. Extensive numerical testing has also shown that the convergence properties of the method are sensitive to the values chosen for the other parameters in the penalty function. It would be of great benefit to develop a general guide as to the choice of values of these parameters, especially considering the vast array of applications of the exact penalty approach [6, 15, 49, 129–131]. A deeper understanding of the dependence of the penalty function on these parameters in required. Finally, it would be interesting to compare the performance of the exact penalty method to other existing optimization methods.

Since we have assumed that a solution exists for each of the classes of problem solved in this thesis, it would also be useful to explore the conditions under which a solution exists for each class.

Bibliography

[1] P. Antsaklis, "Special issue on hybrid systems: theory and applications. a brief introduction to the theory and applications of hybrid systems", *Proceedings of the IEEE*, vol. 88, no. 7, pp. 879–887, 2000.

[2] R. E. Bellman, Dynamic Programming. Princeton University Press, Princeton, NJ. Republished 2003: Dover, ISBN 0-486-42809-5, 1957.

[3] R. Bellman, "The theory of dynamic programming", *Bulletin of the American Mathematical Society* vol. 60, no. 6, pp. 503–515, 1954.

[4] R. Bellman and S. Dreyfus, Applied Dynamic Programming, Princeton University Press, Princeton, N.J., 1962.

[5] E. A. Blanchard, R. Loxton and V. Rehbock, "A computational algorithm for a class of non-smooth optimal control problems arising in aquaculture operations" *Journal of Applied Mathematics and Computation*, vol. 219, no. 16, pp. 8738–8746, 2013.

[6] E. Blanchard, R. Loxton and V. Rehbock, "Optimal control of impulsive switched systems with minimum subsystem durations", *Journal of Global Optimization*, DOI:10.1007/s10898-013-0109-3, 2013.

[7] J. M. Blatt, "Optimal control with a cost of switching control", Journal of the Australian Mathematical Society – Series B: Applied Mathematics, vol. 19, pp. 316–332, 1976.

[8] M. Boccadoro, Y. Wardi, M. Egerstedt and E. Verriest, "Optimal control of switching surfaces in hybrid dynamical systems" *Discrete Event Dynamic Systems: Theory and Applications*, vol. 15, pp. 433–448, 2005.

[9] M. Borchardt, "An exact penalty approach for solving a class of minimization problems with boolean variables", *Optimization: A Journal of Mathematical Programming and Operations Research*, vol. 19, no. 6, pp. 829–838, 1988.

[10] J. V. Breakwell, "The optimization of trajectories", *SIAM Journal on Optimization*, vol. 7, no. 2, pp. 215–247, 1959.

[11] A.E. Bryson Jr., "Optimal control–1950 to 1985", *Control Systems IEEE*, vol. 16, no. 3, pp. 26–33, 1996.

[12] M. J. Cánovas, A. Hantoute, J. Parra and F. J. Toledo, "Calmness of the Argmin Mapping in Linear Semi–Infinite Optimization", *Journal of Optimization Theory and Applications*, vol. 160, no. 1, pp. 111–126, 2014.

[13] C. G. Cassandras, D. L. Pepyne and Y. Wardi, "Optimal control of a class of hybrid systems", *IEEE Transactions on Automatic Control*, vol. 46, no. 6, pp. 398–415, 2001.

[14] Q. Chai, R. Loxton, K. L. Teo and C Yang, "A max–min control problem arising in gradient elution chromatography", *Journal of Industrial Engineering Chemistry Research*, vol. 51, no. 17, pp. 6137–6144, 2012.

[15] Q. Chai, R. Loxton, K. L. Teo and C Yang, "A class of optimal state–delay control problems", *Nonlinear Analysis: Real World Applications* vol. 14, pp. 1536–1550, 2013.

[16] Q. Chai, R. Loxton, K. L. Teo and C Yang, "A unified parameter indentification method for nonlinear time-delay systems", *Journal of Industrial and Management Optimization* vol. 9, no. 2, pp. 471–486, 2013.

[17] F. E. Curtis and M. L. Overton, "A sequential quadratic programming algorithm for nonconvex, nonsmooth constrained optimization", *SIAM Journal on Optimization*, vol. 2, no. 22, pp. 474–500, 2012.

[18] Z. G. Feng, K. L. Teo and V. Rehbock, "Optimal Sensor Scheduling in Continuous Time" *Dynamic Systems and Applications* vol. 17, pp. 331–350, 2008.

[19] R. Fletcher, "An exact penalty function for nonlinear programming with inequalities", *Mathematical Programming: North Holland Publishing Company*, vol. 5, no. 1, pp. 129–150, 1973.

[20] M. Gerdts and S. Sagar, "Mixed–integer DAE optimal control problems: Necessary conditions and bounds", *Control and Optimization with Differential–Algebraic Constraints*, (publisher SIAM), pp. 189–212, 2012.

[21] R. Ghosh and C. Tomlin, "Symbolic reachable set computation of piecewise affine hybrid automata and its application to biological modelling: Delta–Notch protein signalling", *Systems Biology*, vol. 1, no. 1, pp. 170–183, 2004.

[22] R. Goebel and R. T. Rockafellar, " Linear–convex control and duality", *Advances in Mathematics for Applied Sciences*, vol. 76, pp. 280–299, 2008.

[23] R. Goebel, R. Sanfelice and A. Teel, "Hybrid dynamical systems", *Control Systems, IEEE*, vol. 29, no. 2, pp. 28—93, 2009.

[24] C. J. Goh and K. L. Teo, "Control Parametrization: a unified approach to optimal control problems with general constraints", *Automatica*, vol. 24, no. 1, pp. 3–18, 1988.

[25] D. Görges, M. Izák and S. Liu, "Optimal control and scheduling of switched systems", *IEEE Transactions on Automatic Control*, vol. 56, no. 1, pp. 135–140, 2011.

[26] W. Grimm and A. Markl, "Adjoint Estimation from a Multiple–Shooting Method", *Journal of Optimization Theory and Applications*, vol. 92, pp. 263–283, 1997.

[27] W. Hager, "Runge-kutta methods in optimal control and the transformed adjoint system", *Numerische Mathematik*, vol. 87, no. 2, pp. 247–282, 2000.

[28] C. R, Hargraves and S. W. Paris, "Direct trajectory optimization using nonlinear programming and collocation", *Journal of Guidance, Control and Dynamics*, vol. 10, no. 4, pp. 338–342, 1987.

[29] L. Hasdorff, Gradient optimization and nonlinear control. New York: John Wiley, 1976.

[30] S. Hedlund and A. Rantzer, "Optimal control of hybrid systems", *Decision and proceedings of the 38th IEEE conference*, vol. 4, pp. 3972–3977, 1999.

[31] C. Y. F. Ho, B. W. K. Ling, Y. Q. Liu, P. K. S. Tam and K. L. Teo, "Optimal PWM control of switched capacitor DC–DC power converters via model transformation and enhancing control techniques", *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 55, no. 5, pp. 1382–1391, 2008.

[32] W. Huyer and A. Neumaier, "A new exact penalty function", *SIAM Journal on Optimization*, vol. 13, no. 4, pp. 1141–1158, 2003.

[33] C. Hvingel and M. C. S. Kingsley, "A framework to model shrimp stock dynamics and to quantify the risk associated with alternative management options, using Bayesian methods", *ICES Journal of Marine Science*, vol. 63, no. 1, pp. 68–82, 2006.

[34] L. S. Jennings, M. E. Fisher, K. L. Teo and C. J. Goh, MISER 3.3 Optimal Control Software: Theory and User Manual, University of Western Australia, Perth, 2004.

[35] L. S. Jennings and K. L. Teo, "A Numerical algorithm for constrained optimal control problems with applications to harvesting", *Dynamics of Complex Interconnected Biological Systems: Mathematical Modelling* vol. 6, pp. 218–234, 1990.

[36] C. Jiang, Q. Lin, C. Yu, K.L. Teo, G.R. Duan, "An exact penalty method for free terminal time optimal control problem with continuous inequality constraints", *Optimization Thoery and Applications*, vol. 154, pp. 30–53, 2012.

[37] C. Jiang, K. L. Teo, R. Loxton and G. R. Duan, "A neighbouring extremal solution for an optimal switched impulsive control problem", *Journal of Industrial and Management Optimization*, vol. 8, pp. 591–609, 2012.

[38] C. Y. Kaya and J. M. Martínez, "Euler discretization and inexact restoration for optimal control", *Journal of Optimization Theory and Applications*, vol. 134, no. 2, pp. 191—206, 2007.

[39] T. V. Keulen, J. Gillot, B. D. Jagar and M. Steinbuch, "Solution for state constrained optimal control problems applied to power split control for hybrid vehicles", *Automatica*, vol. 50, pp. 187–192, 2014.

[40] A. Khadra, X. Liu and X. Shen, "Application of impulsive syncronization to communication security" *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, **50**, no. 3, pp. 341–351, 2003.

[41] H. W. Kuhn, "Nonlinear programming: A historical view", *Traces and Emergence of Nonlinear Programming*, DOI:10.1007/978-3-0348-0439-4-18, 2014.

[42] H. W. Kuhn and A. W. Tucker, "Nonlinear programming", *Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability*, University of California Press, Berkeley, California, pp. 481–492, 1951.

[43] H. W. J. Lee and K. L. Teo, "Control parametrization enhancing technique for solving a special ODE class with state dependent switch", *Journal of Optimization Theory and Applications*, vol. 118, no. 1, pp. 55–66, 2003.

[44] H. W. J. Lee, K. L. Teo and A. E. B. Lim, "Sensor scheduling in continuous time problems", *Automatica*, vol. 37, pp. 2017–2023, 2001.

[45] H. W. J. Lee, K. L. Teo, V. Rehbock and L. S. Jennings, "Control parameterization enhancing technique for time optimal control problems", *Dynamic Systems and Applications*, vol. 6, no. 2, pp. 243–262, 1997.

[46] H. W. J. Lee, K. L. Teo, V. Rehbock and L. S. Jennings, "Control parameterization enhancing technique for time optimal discrete-valued control problems", *Automatica*, vol. 35, no. 8, pp. 1401–1407, 1999.

[47] S. J. Leese, "Convergence of gradient methods for optimal control problems", *Journal of Optimization Theory and Applications*, vol. 21, no. 3, pp. 329–337, 1977.

[48] A. Li, E. Feng and L. Wang, "Impulsive optimal control model for the trajectory of horizontal wells", *Journal of Computational and Applied Mathematics* vol. 223, no. 2, pp. 893–900, 2009.

[49] B. Li, C. J. Yu, K. L. Teo and G. R. Duan, "An exact penalty function method for continuous inequality constrained optimal control problems", *Journal of Optimization Theory and Application* vol. 151, no. 2, pp. 260–291, 2011.

[50] R. Li and Y. Shi, "Finite-time optimal consensus control for second-order multi-agent systems", *Journal of Industrial and Management Optimization*, vol. 10, no. 3, pp. 929–943, 2014.

[51] R. Li, K. L. Teo, K. H. Wong and G. R. Duan, "Control parameterization enhancing transform for optimal control of switched systems", *Mathematical and Computer Modelling*, vol. 43, no. 11–12, pp. 1393–1403, 2006.

[52] Q. Lin, R. Loxton and K. L. Teo, "The control parameterization method for nonlinear optimal control: A Survey", *Journal of Industrial and Management Optimization*, vol. 10, no. 1, pp. 275–309, 2014.

[53] Q. Lin, R. Loxton, K. L. Teo and Y. H. Wu, "A new computational method for optimizing nonlinear impulsive systems", *Dynamics of Continuous Discrete and Impulsive Systems: Series B – Applications and Algorithms*, vol. 18, pp. 59–76, 2011.

[54] Q. Lin, R. Loxton, K.L. Teo, Y.H. Wu, "A new computational method for a class of free terminal time optimal control problems", *Pacific Journal of Optimization*, vol. 7, pp. 63–81, 2011.

[55] Q. Lin, R. Loxton, K. L. Teo, Y. H. Wu and C. Yu "A new exact penalty method for semi–infinite programming problems", *Journal of Computational and Applied Mathematics*, vol. 261, pp. 271–286, 2014.

[56] B. W. K. Ling, C. Z. Wu, K. L. Teo and V. Rehbock, "Global optimal design of IIR filters via constraint transcription and filled function methods", *Journal of Circuits, Systems and Signal Processing*, vol. 32, no. 3, pp. 1313–1334, 2013.

[57] C. Liu, Z. Gong, E. Feng and H. Yin, "Modelling and optimal control for nonlinear multi-stage dynamical system of microbial fed-batch culture" *Journal of Industrial and Management Optimization*, vol. 5, no. 4, pp. 835–850, 2009.

[58] C. Liu, Z. Gong, B. Shen and E. Feng, "Modelling and optimal control for a fed-batch fermentation process" *Applied Mathematical Modelling*, vol. 37, pp. 695–706, 2013.

[59] Y. Liu, A. Eberhard and K. L. Teo, "A numerical method for a class of mixed switching and impulsive optimal control problems", *Computers and Mathematics with Applications*, vol. 52, pp. 625–636, 2006.

[60] Y. Liu, K. L. Teo, L. S. Jennings and S. Wang, "On a class of optimal control problems with state jumps", *Journal of Optimization Theory and Applications*, vol. 98, pp. 65–82, 1998.

[61] Y. Liu and K. L. Teo, "Computational method for a class of optimal switching control problems", *Progress in Optimization: Applied Optimization*, vol. 39, pp. 221–237, 2000.

[62] K. Lorenzen, Population Dynamics in Aquaculture Compendium, CAB International, Wallingford, UK, 2006.

[63] R. Loxton, Q. Lin and K. L. Teo, "Switching time optimization for nonlinear switched systems: direct optimization and the time-scaling transformation", (Submitted).

[64] R. Loxton, Q. Lin, V. Rehbock and K. L. Teo, "Control parameterization for optimal control problems with continuous inequality constraints: New convergence results", *Numerical Algebra, Control and Optimization*, vol. 2, no. 3, pp. 571–599, 2012.

[65] R. Loxton, K. L. Teo and V. Rehbock, "Optimal control problems with multiple characteristic time points in the objective and constraints", *Automatica*, vol. 44, pp. 2923–2929, 2008.

[66] R. Loxton, K. L. Teo and V. Rehbock, "Computational method for a class of switched system optimal control problems", *IEEE Transactions on Automatic Control*, vol. 54, pp. 2455–2460, 2009.

[67] R. C. Loxton, K. L. Teo, V. Rehbock and W. K. Ling, "Optimal switching instants for a switched-capacitor DC/DC power converter", *Automatica*, vol. 45, pp. 973–980, 2009.

[68] R. C. Loxton, K. L. Teo, V. Rehbock and K. F. C. Yiu, "Optimal control problems with a continuous inequality constraint on the state and the control", *Automatica*, vol. 45, no. 10, pp. 2250–2257, 2009.

[69] S. Lucidi and F. Rinaldi, "Exact penalty functions for nonlinear integer programming problems", *Journal of Optimization Theory and Applications*, vol. 145, no. 3, pp. 479–488, 2010.

[70] D. G. Luenberger and Y. Ye, Linear and Nonlinear Programming, 3rd edition, New York: Springer 2008.

[71] R. B. Martin, "Optimal control drug scheduling of cancer chemotherapy", *Automatica*, vol. 28, no. 6, pp. 1113–1123, 1992.

[72] R. B. Martin and K. L. Teo, Optimal Control of Drug administration in Cancer Chemotherapy, Singapore: World Scientific, 1994.

[73] E. Mojica Nava, R. Meziat, N. Quijano, A. Gauthier and N. Rakoto–Ravalontsalama, "Optimal control of switched systems: A polynomial approach", *IFAC Proceedings Volumes*, vol. 17, no. 1, pp. 7808–7813, 2008.

[74] B. S. Mordukhovich, R. T. Rockafellar and M. E. Sarabi, "Characterizations of full stability in constrained optimization", *SIAM Journal on Optimization*, vol. 23, no. 3, pp. 1810–1849, 2013.

[75] D. D. Morrison, J. D. Riley and J. F. Zancanaro, "Multiple shooting method for two-point boundary value problems", *Communications of the ACM*, vol. 5, no. 12, pp. 613–614, 1962.

[76] W. Murray and K. M. Ng, "An algorithm for non-linear optimization problems with binary variables", *Computational Optimization and Applications*, vol. 47, no. 2, 257–288, 2010.

[77] J. Nocedal and S. J. Wright, Numerical Optimization, 2nd edition, New York: Springer 2006.

[78] H. J. Oberle, "Numerical solution of minimax optimal control problems by multiple shooting technique", *Journal of Optimization Control and Application*, vol. 50, no. 2, pp. 331–357, 1986.

[79] B. Passenberg, *Theory and Algorithms for Indirect Methods in Optimal Control of Hybrid Systems*. Ph.D. Thesis, Institute of Automatic Control Engineering, LSR, München, Germany 2012.

[80] Y. Peng, X. Xiang, "Second order nonlinear impulsive time-variant systems with unbounded perturbation and optimal controls", *Journal of Industrial and Management Optimization*, vol. 4, pp. 17–32, 2008.

[81] E. Polak, "A historical survey of computational methods in optimal control", *SIAM Review*, vol. 15, no. 2, 553–584, 2006.

[82] L. S. Pontryagin, Mathematical Theory of Optimal Processes. New York: John Wiley and Sons, 1962.

[83] A. V. Rao, "A survey of numerical methods for optimal control", *AAS/AIAA Astrodynamics specialist conference* AAS Paper 09–334, Pittsburgh, PA, 2009.

[84] V. Rehbock, *Tracking Control and Optimal Control*, Ph.D. Thesis, University of Western Australia, Perth, 1994.

[85] V. Rehbock, K. L. Teo, L. S. Jennings and H. W. J. Lee, "A survey of the control parametrization and control parametrization enhancing methods for constrained optimal control problems", *Progress in Optimization: Contributions from Australasia*, vol. 30, pp. 247–275, 1999.

[86] G. Rivera–Velázquez, L. A. Soto, I. H. Salgado–Ugarte and E. J. Naranjo, "Growth, mortality and migratory pattern of white shrimp in the Carretas–Pereyra coastal lagoon system, Mexico", *International Journal of Tropical Biology*, vol. 56, no. 2, pp. 523–533, 2008.

[87] R. T. Rockafellar, "Duality and stability in extremum problems involving convex functions", *Pacific Journal of Mathematics*, vol. 21, pp. 167–187, 1967.

[88] R. T. Rockafellar, "Convex Analysis", vol. 28 of the Princeton Mathematics Series, Princeton University Press, 1970.

[89] R. T. Rockafellar, "Existence theorems for general control problems of Bolza and Lagrange", *Advances in Mathematics*, vol. 15, no. 3, pp. 312–333, 1975.

[90] R. T. Rockafellar, "Directional differentiability of the optimal value function in a nonlinear programming problem", *Mathematical Programming Study*, vol. 21, pp. 213–226, 1984.

[91] R. T. Rockafellar, "Equivalent subgradient versions of Hamiltonian and Euler–Lagrange equations in variational analysis", *SIAM Journal on Control and Optimization*, vol. 34, no. 4, pp. 1300–1315, 1996.

[92] R. T. Rockafellar, "Hamilton–Jacobi theory and parametric analysis in fully convex problems of optimal control", *Journal of Global Optimization*, vol. 28, pp. 419–431, 2004.

[93] T. Ruby, V. Rehbock and W. B. Lawrance, "Optimal control of hybrid power systems", *Dynamics of Continuous, Discrete and Impulsive Systems: Series B - Applications and Algorithms*, vol. 10, pp. 429–439, 2003.

[94] S. Sagar, *Numerical Methods for Mixed Integer Optimal Control Problems*, Ph.D. Thesis, Universität Heidelberg, Heidelberg, Germany, 2005.

[95] T. Sakamoto, N. Hori, Y. Ochi, "Exact linearization and discretization of nonlinear systems satisfying a Lagrange PDE condition", *Transactions of the Canadian Society for Mechanical Engineering*, vol. 35, no. 2, pp. 215–228, 2011.

[96] K. Schittkowski, "Numerical solution of systems of nonlinear inequalities", *Optimization and Operations Research, Lecture Notes in Economics and Mathematical Systems*, vol. 117, pp. 259–272, 1976.

[97] K. Schittkowski, "An adaptive precision method for nonlinear optimization problems", *SIAM Journal on Control and Optimization*, vol. 17, no. 1, 82–98, 1979.

[98] K. Schittkowski, "NLPQL: A fortran subroutine solving constrained nonlinear programming problems", *Annals of Operations Research*, vol. 5, pp. 485–500, 1986.

[99] K. Schittkowski, "NLPQLP: A fortran implementation of a sequential quadratic programming algorithm with distributed and non-monotone line search", *User's Guide Version 2.0, University of Bayreuth*, 2004.

[100] K. Schittkowski, "A collection of 186 test problems for nonlinear mixed–integer programming", *Report, Department of Computer Science, University of Bayreuth*, 2012.

[101] K. Schittkowski, "A robust implementation of a sequential quadratic programming algorithm with successive error restoration", *Optimization Letters*, vol. 5, pp. 283–296, 2011.

[102] K. Schittkowski and Y. Yuan, "Sequential quadratic programming methods", *Wiley Encyclopedia of Operations Research and Management Science*, 2011.

[103] A. L. Schwartz and E. Polak, "Consistent approximations for optimal control problems Based on Runge–Kutta integration, *SIAM Journal on Optimization and Control*, vol. 34 no. 4, pp. 1235–1269, 1996.

[104] D. K. Shin, Z. Gürdal and O. H. Griffin Jr., "A penalty approach for nonlinear optimization with discrete design variables", *Engineering Optimization*, vol. 16, no. 1, pp. 29–42, 1990.

[105] M. D. Stuber, A. Wechsung, A. Sundaramoorthy and P. I. Barton, "Worst–case design of subsea production facilities using semi–infinite programming", *AIChE Journal*, DOI: 10.1002/aic.14447, 2014.

[106] O. Stursberg and S. Engell, "Optimal control of switched continuous systems using mixed-integer programming", *15th IFAC World Congress, Barcelona, Spain*, Paper Th-A06-4, pp. 209–210, 2002.

[107] O. Stursberg and S. Panek, "Control of switched hybrid systems based on disjunctive formulations", *Hybrid Systems: Computation and Control*, vol. 2289, pp. 421–435, 2002.

[108] H. J. Sussmann, "A maximum principle for hybrid optimal control problems" *38th IEEE conference on Decision and Control*, vol. 4, pp. 425-430, 1999.

[109] H. J. Sussmann and J. C. Willems, "300 Years of optimal control: from the Brachystochrone to the Maximum Principle" *IEEE Control Systems*, (Phoenix, AZ, Dec. 7-10), pp. 435–450, 1997.

[110] K. Teo and C. Goh, "A simple computational procedure for optimization problems with functional inequality constraints", *IEEE Transactions on Automatic Control*, vol. 32, no. 10, pp. 940--941, 1987.

[111] K. L. Teo, C. J. Goh and K. H. Wong, A unified computational approach to optimal control problems. Longman Scientific and Technical: Essex, 1991.

[112] K. L. Teo and L. S. Jennings, "Nonlinear optimal control problems with continuous state inequality constraints", *Journal of Optimization Theory and Application* vol. 63, no. 1, pp. 1–22, 1989.

[113] K. L. Teo, L. S. Jennings, H. W. J. Lee and V. Rehbock, "The control parameterization enhancing transform for constrained optimal control problems", *The Journal of the Australian Mathematical Society. Series B. Applied Mathematics*, vol. 40, no. 3, pp. 314–335, 1999.

[114] K. L. Teo, L. S. Jennings, H. W. J. Lee and V. Rehbock, "Numerical solution of an optimal control problem with variable time points in the objective function", *The ANZIAM Journal*, vol. 43, no. 4, pp. 463–478, 2002.

[115] X. Tian, P. Leung and E. Hochman, Shrimp growth functions and their economic implications. *Aquacultural Engineering*, vol. 12, no. 2, pp. 81–96, 1993.

[116] S. Wang, K. L. Teo and H. W. J. Lee, A new approach to nonlinear mixed discrete programming problems, *Engineering Optimization*, vol. 30, no. 3, pp. 249–262, 1998.

[117] Y. Wang, X. Shi, Z. Zuo, M. Z. Q. Chen and Y. Shao, "On finite–time stability for nonlinear impulsive switched systems", *Nonlinear Analysis: Real World Applications*, vol. 14, no. 1, pp. 807–814, 2013.

[118] S. Wei, K. Uthaichana, M. Zefran, R. A. DeCarlo and S. Bengea, "Applications of numerical optimal control to nonlinear hybrid systems", *Nonlinear Analysis: Hybrid Systems*, vol. 1, no. 2, pp. 264–279, 2007.

[119] W. Wei, K. L. Teo and Z. Zhan, "A numerical method for an optimal control problem with minimum sensitivity on coefficient variation", *Applied Mathematics and Computation*, vol. 218, pp. 1180–1190, 2011.

[120] K. H. Wong, L. S. Jennings and F. Benyah, "The control parameterization enhancing transform for constrained time–delayed optimal control problems", *The ANZIAM Journal*, vol. 43, pp. E154–E185, 2002.

[121] S. Woon, R. Loxton and V. Rehbock, "A modified discrete filled function algorithm for solving nonlinear discrete optimization problems", Proceedings of the 5th International Conference on Optimization and Control with Applications, Beijing, China, December 4–8, 2012.

[122] C. Z. Wu and K. L. Teo, "Global impulsive optimal control computation", *Journal of Industrial and Management Optimization*, vol. 2, pp. 435–450, 2006.

[123] X. Wu, K. Zhang and C. Sun, "Numerical algorithm for a class of constrained optimal control problems of switched systems", *Journal of Numerical Algorithms*, DOI: 10.1007/s11075-013-9822-8, 2014.

[124] H. Xu and K. L. Teo, "H_∞ optimal stabilization of a class of uncertain impulsive systems: An LMI approach", *Journal of Industrial and Management Optimization*, vol. 5, pp. 153–159, 2009.

[125] X. Xu and P. J. Antsaklis, "Results and perspectives on computational methods for optimal control of switched systems", *Hybrid Systems: Computation and Control Lecture Notes in Computer Science*, vol. 2623, pp. 540—555, 2003.

[126] X. P. Xu and P. J. Antsaklis, "Optimal control of switched systems based on parameterization of the switching instants", *IEEE Transactions on Automatic Control*, vol. 49, pp. 2–16, 2004.

[127] Y. Yang, Z. Wu and F. Bai, "A filled function method for constrained nonlinear integer programming", *Journal of Industrial and Management Optimization*, vol. 4, pp. 353–362, 2008.

[128] J. X. Ye, Y. D. Zhang, E. M. Feng, Z. L. Xiu and H. C. Yin, "Nonlinear hybrid system and parameter identification of microbial fed-batch culture with open loop glycerol input and pH logic control", *Applied Mathematical Modelling*, vol. 36, pp. 357–369, 2012.

[129] C. Yu, K. L. Teo and Y. Bai, "An exact penalty function method for nonlinear mixed discrete programming problems", *Optimization Letters*, vol. 7, no. 1, pp. 23–38, 2013.

[130] C. Yu, K. L. Teo, L. Zhang and Y. Bai, "A new exact penalty function method for continuous inequality constrained optimization problems", *Journal of Industrial and Management Optimization*, vol. 6, no. 4, pp. 895–910, 2010.

[131] C. Yu, B. Li, R. Loxton and K. L. Teo, "Optimal discrete-valued control computation", *Journal of Global Optimization*, vol. 56, no. 2, pp. 503–518, 2013.

[132] C. Yu, K. L. Teo, L. Zhang and Y. Bai, "On a refinement of the convergence analysis for the new exact penalty function method for continuous inequality constrained optimization problem", *Journal of Industrial and Management Optimization*, vol. 8, no. 2, pp. 485–491, 2012.

[133] R. Yu and P. Leung, "Optimal partial harvesting schedule for aquaculture operations", *Marine Resource Economics*, vol. 21, pp. 301–315, 2006.

[134] R. Yu, P. Leung and P. Bienfang, "Modelling partial harvesting in intensive shrimp culture: A network-flow approach", *European Journal of Operational Research*, vol. 193, pp. 262–271, 2009.

[135] F. Zhu and P. J. Antsaklis, "Optimal control of hybrid switched systems: A brief survey", *Discrete Event Dynamic Systems*, DOI 10.1007/s10626-014-0187-5, 2014.